



**TUGAS AKHIR - KI141502**

# **PENENTUAN HARGA DENGAN METODE BACK PROPAGATION PADA APLIKASI E-COMMERCE CARIKOS BERBASIS WEB**

**RIFQI NUR FADHILAH**  
**NRP 5113100170**

Dosen Pembimbing I  
Dr.tech. Ir. Raden Venantius Hari Ginardi, M.Sc.

Departemen Teknik Informatika  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017



**TUGAS AKHIR - KI141502**

# **PENENTUAN HARGA DENGAN METODE BACK PROPAGATION PADA APLIKASI E-COMMERCE CARIKOS BERBASIS WEB**

**RIFQI NUR FADHILAH  
NRP 5113100170**

**Dosen Pembimbing I  
Dr.tech. Ir. Raden Venantius Hari Ginardi, M.Sc.**

**Departemen Teknik Informatika  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017**



**UNDERGRADUATE THESES - KI141502**

**COST ESTABLISHMENT WITH BACK  
PROPAGATION METHOD ON CARIKOS WEB  
BASED APPLICATION E-COMMERCE**

**Rifqi Nur Fadhilah  
NRP 5113100170**

**First Advisor  
Dr.tech. Ir. Raden Venantius Hari Ginardi, M.Sc.**

**Department of Informatics  
Faculty of Information Technology  
Sepuluh Nopember Institute of Technology  
Surabaya 2017**

*[Halaman ini sengaja dikosongkan]*

## **LEMBAR PENGESAHAN**

### **PENENTUAN HARGA MENGGUNAKAN BACK PROPAGATION PADA APLIKASI E-COMMERCE CARIKOS BERBASIS WEB**

#### **TUGAS AKHIR**

Diajukan Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Bidang Studi Manajemen Informasi  
Program Studi S-1 Departemen Teknik Informatika  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember

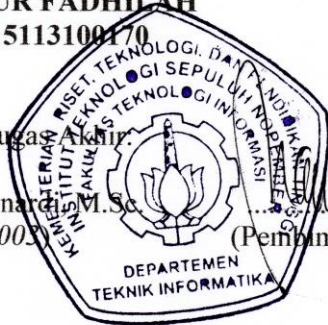
Oleh:

**RIFQI NUR FADHILAH**

**NRP: 5113100170**

Disetujui oleh Pembimbing Tugas Akhir

1. Dr.tech. Ir. R. V. Hari Ginardi, M.Sc.  
(NIP. 196505181992031003) (Pembimbing 1)



**SURABAYA**

**Juli, 2017**

*[Halaman ini sengaja dikosongkan]*

# **PENENTUAN HARGA MENGGUNAKAN METODE BACK PROPAGATION PADA APLIKASI E-COMMERCE CARIKOS BERBASIS WEB**

**Nama Mahasiswa : RIFQI NUR FADHILAH**  
**NRP : 5113100170**  
**Jurusan : Departemen Informatika FTIF-ITS**  
**Dosen Pembimbing 1 : Dr. tech. Ir. Raden Venantius Hari  
Ginardi, M.Sc.**

## **Abstrak**

*Aplikasi CariKos merupakan aplikasi e-commerce berbasis web yang mewadahi para pencari indekos agar mudah dalam mencari indekos. Aplikasi ini menyajikan harga yang fluktuatif sesuai dengan permintaan pasar. Untuk mendapatkan harga yang fluktuatif ini digunakan metode back propagation untuk menentukan harga. Penentuan harga ini dipengaruhi oleh banyak atau sedikitnya permintaan pasar akan indekos beserta nilai skor indekos.*

*Uji coba yang dilakukan sebanyak tiga macam uji coba, yaitu uji coba back propagation, uji coba fungsionalitas, dan uji coba tingkat kepuasan serta pengukuran keuntungan/kerugian. Uji coba metode back propagation yang digunakan untuk menentukan harga menghasilkan nilai rata-rata MSE sebesar 0.030167. Uji coba fungsionalitas dan uji coba tingkat kepuasan serta pengukur keuntungan atau kerugian dilakukan oleh 5 partisipan. Uji coba fungsionalitas ini dilakukan untuk mengukur keberhasilan dalam menjalani proses bisnis. Pada uji coba fungsionalitas, semua fungsionalitas dalam aplikasi berjalan dengan baik dan benar. Uji coba tingkat kepuasan serta pengukur keuntungan atau kerugian dilakukan untuk mengukur tingkat kepuasan partisipan dengan harga yang ditampilkan dan untuk mengetahui keuntungan/kerugian yang didapat. Tingkat kepuasan partisipan*

*terhadap harga yang ditampilkan pada aplikasi sangat baik. Lima dari lima partisipan merasa puas dengan harga yang ditampilkan pada aplikasi CariKos. Aplikasi CariKos mendapatkan keuntungan dari pemesanan yang dilakukan oleh lima partisipan sebanyak tiga kali pemesanan dalam satu periode. Sehingga metode backpropagation cocok untuk melakukan penentuan harga indekos.*

***Kata kunci: Back Propagation, E-Commerce, Fluktuatif, MSE , Konvensional, Teknologi Informasi***



# **COST ESTABLISHMENT WITH BACK PROPAGATION METHOD ON CARIKOS WEB BASED APPLICATION E- COMMERCE**

**Student's Name : RIFQI NUR FADHILAH**  
**Student's ID : 5113100170**  
**Department : Departement of Informatics FTIF-ITS**  
**First Advisor : Dr. tech. Ir. Raden Venantius Hari  
Ginardi, M.Sc**

## **Abstract**

*The CariKos app is a web-based e-commerce application make to easier boarding house seeker to search boarding house. This application presents a fluctuating price in accordance with market demand. To get the fluctuating price is used back propagation method to determine the price. This pricing is influenced by the large or small market demand for boarding house and the value of the boarding house score .*

*Testing in three different types of test, test back propagation, test functionality, and test the level of satisfaction and measurement of profit/loss. The test of the back propagation method to determine the price result an average MSE is 0.030167. Test of functionality and test of satisfaction level and measurement profit or loss by five participants. This functionality test is performed to measure success in business processes. On functionality test, all functionality in the application runs correctly. Test of satisfaction level and measurement profit/loss is to measure the level of satisfaction of participants with the displayed price on application and to know the profit or loss be obtained. The level of participant satisfaction with the price shown on the application is very good. Five of the five participants were satisfied with the price shown in the CariKos application. The CariKos application benefits from booking made by five participants for three orders in*

*one period. So the back propagation method is suitable for determining the price of the boarding house.*

***Keywords : Back Propagation, Conventional, E-Commerce, Fluctuating, Information Technology, MSE***

## KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillahirabbil'alamin, segala puji bagi Allah SWT yang telah melimpahkan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul:

### **“PENENTUAN HARGA DENGAN METODE BACK PROPAGATION PADA APLIKASI E-COMMERCE CARIKOS BERBASIS WEB”**

Terselesaikannya Tugas Akhir ini tidak terlepas dari bantuan dan dukungan banyak pihak, Oleh karena itu melalui lembar ini penulis ingin mengucapkan terima kasih dan penghormatan kepada:

1. Allah SWT serta junjungan Nabi Muhammad SAW, karena limpahan rahmat dan karunia-Nya penulis dapat menyelesaikan Tugas Akhir dan juga perkuliahan di Teknik Informatika ITS.
2. Ayah dan Ibu penulis, Joni Sujono dan Yulianti yang tiada hentinya memberikan dukungan doa, moral, dan material kepada penulis sehingga penulis dapat menyelesaikan Tugas Akhir ini.
3. Adik penulis, Alya Sabrina Faradilla dan Farrel Ardhan Fabiano serta keluarga tercinta yang telah memberikan dukungan dan semangat kepada penulis.
4. Bapak Dr. tech. Ir. Raden Venantius Hari Ginardi selaku pembimbing I dan dosen wali di semester 8 yang telah membimbing dan memberikan motivasi, nasihat dan bimbingan dalam menyelesaikan Tugas Akhir ini.
5. Royyana Muslim Ijtihadie, S.Kom., M.Kom., Ph.D. selaku dosen wali dari semester 1 – 7 penulis yang telah

memberikan arahan kepada penulis selama menjalani perkuliahan di Teknik Informatika ITS.

6. Lian Ardelia yang merupakan orang yang selalu mendukung, memberikan semangat, dan selalu mengingatkan untuk bangkit ketika merasa gagal.
7. Astidhita Nuraini Latifah dan Luwandino Wismar sebagai teman sekaligus sahabat satu kelompok dalam pengembangan aplikasi CariKos ini.
8. Zaza, Nindy, Saddam, Hariyanto, dan Apip yang merupakan sahabat penulis yang selalu membantu, menghibur, dan bermain bersama penulis.
9. Nyoman, Arvi, Ridho, Aqsha, Mooy, Kevin, dan Bagus sebagai teman sepermainan yang selalu menghibur, bersenang-senang bersama dan selalu mengingatkan penulis.
10. Luthfa, Nikitta, dan Seta, sahabat SMA penulis yang selalu memberikan motivasi, semangat, dan doa.
11. Serta semua pihak yang telah turut membantu penulis dalam menyelesaikan Tugas Akhir ini.

Penulis menyadari bahwa laporan Tugas Akhir ini masih memiliki banyak kekurangan. Oleh karena itu dengan segala kerendahan hati penulis mengharapkan kritik dan saran dari pembaca untuk perbaikan penulis kedepannya. Selain itu, penulis berharap laporan Tugas Akhir ini dapat berguna bagi pembaca secara umum.

Surabaya, Juli 2017

## DAFTAR ISI

LEMBAR PENGESAHAN.....	v
Abstrak .....	vii
Abstract .....	ix
KATA PENGANTAR.....	xi
DAFTAR ISI .....	xiii
DAFTAR TABEL .....	xxi
DAFTAR KODE SUMBER .....	xxiii
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Permasalahan .....	2
1.4 Tujuan .....	3
1.5 Manfaat.....	3
1.6 Metodologi .....	3
1.6.1 Penyebaran Kuesioner .....	3
1.6.2 Studi Literatur .....	4
1.6.3 Analisis dan Desain Perangkat Lunak .....	4
1.6.4 Implementasi Perangkat Lunak.....	5
1.6.5 Pengujian dan Evaluasi.....	5
1.7 Sistematika Penulisan.....	5
BAB II TINJAUAN PUSTAKA.....	7
2.1 Aplikasi Sejenis.....	7
2.2 <i>E - Commerce</i> .....	8
2.2.1 Business to Business (B2B).....	9
2.2.2 <i>Business to Consumer</i> (B2C).....	9
2.2.3 <i>Consumer to Consumer</i> (C2C) .....	10
2.2.4 <i>Consumer to Business</i> (C2B).....	10
2.2.5 <i>Business to Administration</i> (B2A) .....	10
2.2.6 <i>Consumen to Administration</i> (C2A).....	10
2.3 Teori Permintaan .....	11
2.4 Back Propagation Neural Network.....	12
2.5 <i>CodeIgniter</i> .....	14
2.6 Sublime Text .....	16

2.7	Hypertext Preprocessor (PHP)	16
2.8	TCPDF	17
BAB III ANALISIS DAN PERANCANGAN SISTEM		19
3.1	Analisis	19
3.1.1	Analisis Permasalahan	19
3.1.2	Deskripsi Umum Sistem	20
3.1.3	Analisis Kebutuhan Sistem	23
3.1.4	Analisis Aktor	24
3.1.5	Kasus Penggunaan	24
3.1.5.1	Melakukan Pencari Indekos (UC-0001)	26
3.1.5.2	Melakukan Pemesanan Indekos (UC-0002)	27
3.1.5.3	Melakukan Pembatalan Pemesanan Indekos (UC-0003)	29
3.1.5.4	Melakukan Konfirmasi Pembayaran Indekos (UC-0004)	31
3.1.5.5	Mendapatkan <i>E-Ticket</i> (UC-0005)	33
3.1.5.6	Melihat Informasi Indekos (UC-0006)	34
3.1.5.7	Melihat Transaksi (UC-0007)	36
3.1.5.8	Verifikasi Pembayaran (UC-0008)	37
3.1.5.9	Melihat Laporan Keuangan (UC-0009)	38
3.1.5.10	Melihat Pemesanan (UC-0010)	40
3.2	Perancangan Sistem	41
3.2.1	Perancangan Basis Data	41
3.2.2	Perancangan Algoritma	44
3.2.2.1	Perancangan Algoritma Modul Pemesanan	44
3.2.2.2	Perancangan Algoritma Modul Penentuan Harga	45
3.2.3	Perancangan Tampilan Antarmuka	51
3.2.3.1	Perancangan Halaman Beranda	51
3.2.3.2	Perancangan Halaman Hasil Pencarian	52
3.2.3.3	.....Perancangan Halaman Detail Kamar dan Indekos	52
3.2.3.4	Perancangan Halaman <i>Form</i> Pemesanan Kamar	53
3.2.3.5	Perancangan Halaman Detail Pemesanan	54

3.2.3.6	Perancangan Halaman Pembayaran .....	55
3.2.3.7	Perancangan Halaman Konfirmasi Pembayaran .....	56
3.2.3.8	Perancangan Halaman Daftar Transaksi di sisi Pencari Indekos .....	57
3.2.3.9	Perancangan Halaman Utama Admin .....	58
3.2.3.10	Perancangan Halaman Indekos.....	58
3.2.3.11	Perancangan Halaman Pemesanan .....	59
3.2.3.12	Perancangan Halaman Transaksi.....	60
3.2.3.13	Perancangan Halaman Laporan Keuangan..	61
BAB IV IMPLEMENTASI.....		63
4.1	Lingkungan Implementasi.....	63
4.1.1	Lingkungan Implementasi Perangkat Keras .....	63
4.1.2	Lingkungan Implementasi Perangkat Lunak .....	63
4.2	Implementasi Tampilan Antarmuka .....	64
4.2.1	Implementasi Halaman Beranda .....	64
4.2.2	Implementasi Halaman Hasil Pencarian .....	65
4.2.3	Implementasi Halaman Detail Kamar dan Indekos .....	66
4.2.4	Implementasi Halaman <i>Form</i> Pemesanan Kamar... ..	68
4.2.5	Implementasi Halaman Detail Pemesanan.....	69
4.2.6	Implementasi Halaman Pembayaran.....	70
4.2.7	Implementasi Halaman Konfirmasi Pembayaran ...	71
4.2.8	Implementasi Halaman Daftar Transaksi di Sisi Pencari Indekos.....	72
4.2.9	Implementasi Halaman Utama Admin.....	73
4.2.10	Implementasi Halaman Indekos.....	74
4.2.11	Implementasi Halaman Kamar Indekos .....	75
4.2.12	Implementasi Halaman Pemesanan .....	75
4.2.13	Implementasi Halaman Transaksi.....	76
4.2.14	Implementasi Halaman Laporan Keuangan .....	78
4.3	Implementasi Penentuan Harga.....	78
4.3.1	Implementasi <i>Training</i> Data .....	79
4.3.2	Implementasi <i>Testing</i> Data .....	81
4.4	Implementasi Alur Proses Aplikasi .....	83
4.4.1	Implementasi Alur Proses Pencarian .....	83

4.4.2	Implementasi Alur Proses Pemesanan .....	84
4.4.3	Implementasi Alur Proses Pembatalan Pemesanan.....	86
4.4.4	Implementasi Alur Proses Pembayaran .....	87
4.4.5	Implementasi Alur Proses Mendapatkan <i>E-Ticket</i> .....	88
4.4.6	Implementasi Alur Proses Melihat Kamar Indekos .....	91
4.4.7	Implementasi Alur Proses Melihat Indekos .....	92
4.4.8	Implementasi Alur Proses Melihat Transaksi .....	93
4.4.9	Implementasi Alur Proses Verifikasi Pembayaran .....	93
4.4.10	Implementasi Alur Proses Melihat Laporan Keuangan .....	94
4.4.11	Implementasi Alur Proses Melihat Pemesanan.....	95
BAB V Uji COBA DAN EVALUASI.....		97
5.1	Lingkungan Pengujian.....	97
5.2	Skenario Uji Coba .....	97
5.2.1	Skenario Uji Coba <i>Back Propagation</i> .....	98
5.2.2	Skenario Uji Coba Fungsionalitas.....	101
5.2.2.1	Uji Coba Melakukan Pencarian Indekos .....	102
5.2.2.2	Uji Coba Melakukan Pemesanan Indekos .....	102
5.2.2.3	Uji Coba Melakukan Pembatalan Pemesanan Indekos .....	103
5.2.2.4	Uji Coba Melakukan Konfirmasi Pembayaran Indekos .....	104
5.2.2.5	Uji Coba Mendapatkan <i>E-Ticket</i> .....	105
5.2.2.6	Uji Coba Melihat Kamar Indekos.....	106
5.2.2.7	Uji Coba Melihat Indekos.....	107
5.2.2.8	Uji Coba Melihat Transaksi.....	108
5.2.2.9	Uji Coba Verifikasi Pembayaran .....	108
5.2.2.10	Uji Coba Melihat Laporan Keuangan.....	109
5.2.2.11	Uji Coba Melihat Pemesanan .....	110
5.2.3	Skenario Uji Coba Penentuan Harga .....	111
5.2.4	Skenario Uji Coba Tingkat Kepuasan serta Pengukuran Keuntungan atau Kerugian .....	113
5.3	Evaluasi Uji Coba.....	117
5.3.1	Evaluasi Uji Coba <i>Back Propagation</i> .....	117
5.3.2	Evaluasi Uji Coba Fungsionalitas .....	118



5.3.3	Evaluasi Uji Coba Tingkat Kepuasan serta Pengukuran Keuntungan atau Kerugian .....	120
5.3.4	Evaluasi Uji Coba Penentuan Harga.....	121
BAB VI KESIMPULAN DAN SARAN.....		123
6.1	Kesimpulan.....	123
6.2	Saran.....	124
DAFTAR PUSTAKA.....		125
LAMPIRAN .....		127
BIODATA PENULIS.....		129

*[Halaman ini sengaja dikosongkan]*

## DAFTAR GAMBAR

Gambar 1.1 Diagram Permintaan Indekos .....	4
Gambar 2.1 Arsitektur Back Propagation .....	13
Gambar 2.2 Diagram Presentase Penggunaan PHP Framework ..	15
Gambar 3.1 Arsitektur Aplikasi CariKos .....	21
Gambar 3.2 Proses Bisnis Aplikasi CariKos .....	22
Gambar 3.3. Diagram Kasus Penggunaan .....	25
Gambar 3.4. Diagram Aktivitas Melakukan Pencarian Indekos ..	27
Gambar 3.5 Diagram Aktivitas Melakukan Pemesanan Indekos	29
Gambar 3.6 Diagram Aktivitas Melakukan Pembatalan Pemesanan Indekos .....	31
Gambar 3.7 Diagram Aktivitas Melakukan Pembayaran Indekos .....	33
Gambar 3.8 Diagram Aktivitas Mendapatkan E-Ticket.....	34
Gambar 3.9 Diagram Aktivitas Melihat Informasi Indekos .....	35
Gambar 3.10 Diagram Aktivitas Melihat Transaksi.....	37
Gambar 3.11 Diagram Aktivitas Verifikasi Pembayaran .....	38
Gambar 3.12 Diagram Aktivitas Melihat Laporan Keuangan.....	40
Gambar 3.13 Diagram Aktivitas Melihat Pemesanan .....	41
Gambar 3.14 Conceptual Data Model.....	42
Gambar 3.15 Physical Data Model.....	43
Gambar 3.16 Diagram Alir Pemesanan.....	44
Gambar 3.17 Diagram Alir Metode Back Propagation .....	45
Gambar 3.18 Rancangan Halaman Beranda.....	51
Gambar 3.19 Rancangan Halaman Hasil Pencarian.....	52
Gambar 3.20 Rancangan Detail Kamar dan Indekos .....	53
Gambar 3.21 Rancangan Halaman Form Pemesanan Kamar.....	54
Gambar 3.22 Rancangan Halaman Detail Pemesanan .....	55
Gambar 3.23 Rancangan Halaman Pembayaran .....	56
Gambar 3.24 Rancangan Halaman Konfirmasi Pembayaran .....	57
Gambar 3.25 Rancangan Halaman Daftar Transaksi di sisi Pencari Indekos .....	58
Gambar 3.26 Rancangan Halaman Home Admin .....	58
Gambar 3.27 Rancangan Halaman Indekos .....	59

Gambar 3.28 Rancangan Halaman Pemesanan .....	60
Gambar 3.29 Rancangan Halaman Transaksi.....	61
Gambar 3.30 Rancangan Halaman Laporan Keuangan.....	61
Gambar 4.1 Tampilan Navigation Bar Halaman Beranda.....	64
Gambar 4.2 Tampilan Halaman Beranda .....	65
Gambar 4.3 Tampilan Halaman Hasil Pencarian .....	66
Gambar 4.4 Tampilan Halaman Detail Kamar dan Indekos.....	67
Gambar 4.5 Tampilan Halaman Form Pemesanan Kamar .....	68
Gambar 4.6 Tampilan Halaman Detail Pemesanan.....	69
Gambar 4.7 Tampilan Halaman Pembayaran.....	70
Gambar 4.8 Tampilan Halaman Konfirmasi Pembayaran.....	71
Gambar 4.9 Tampilan Halaman Daftar Transaksi di Sisi Pencari Indekos .....	72
Gambar 4.10 Tampilan Halaman Utama Admin.....	73
Gambar 4.11 Tampilan Halaman Indekos .....	74
Gambar 4.12 Tampilan Halaman Kamar Indekos .....	75
Gambar 4.13 Tampilan Halaman Pemesanan.....	76
Gambar 4.14 Tampilan Halaman Transaksi .....	77
Gambar 4.15 Tampilan Halaman Laporan Keuangan .....	78
Gambar 5.1 Grafik Pengujian Iterasi .....	99
Gambar 5.2 Grafik Pengujian Hidden Layer .....	100
Gambar 5.3 Grafik Hasil Pengujian Learning Rate.....	101
Gambar 5.4 Pengujian Konfirmasi Pembayaran Indekos.....	105
Gambar 5.5 Hasil Uji Coba Mendapatkan E-Ticket.....	106
Gambar 5.6 Hasil Uji Coba Verifikasi Pembayaran.....	109
Gambar 5.7 Harga Awal Kamar Deluxe Indekos Terompet .....	111
Gambar 5.8 Harga Akhir Ketika Permintaan Rendah .....	112
Gambar 5.9 Harga Akhir Ketika Permintaan Tinggi.....	112
Gambar 5.10 Grafik Tingkat Kepuasan.....	120
Gambar A.1 Pertanyaan Kuesioner Permintaan Pasar .....	127
Gambar A.2 Pertanyaan Kuesioner Tingkat Kepuasan Terhadap Harga yang Ditampilkan.....	127

## DAFTAR TABEL

Tabel 3.1 Daftar Kebutuhan Sistem .....	23
Tabel 3.2 Daftar Kasus Penggunaan .....	24
Tabel 3.3 Spesifikasi Kasus Penggunaan Melakukan Pencarian Indeks .....	26
Tabel 3.4 Spesifikasi Kasus Penggunaan Melakukan Pemesanan Indeks .....	27
Tabel 3.5 Spesifikasi Kasus Penggunaan Melakukan Pembatalan Pemesanan Indeks .....	30
Tabel 3.6 Spesifikasi Kasus Penggunaan Melakukan Konfirmasi Pembayaran Indeks.....	32
Tabel 3.7 Spesifikasi Kasus Penggunaan Mendapatkan E-Ticket .....	33
Tabel 3.8 Spesifikasi Kasus Penggunaan Melihat Informasi Indeks .....	35
Tabel 3.9 Spesifikasi Kasus Penggunaan Melihat Transaksi .....	36
Tabel 3.10 Spesifikasi Kasus Penggunaan Verifikasi Pembayaran .....	37
Tabel 3.11 Spesifikasi Kasus Penggunaan Melihat Laporan Keuangan.....	39
Tabel 3.12 Spesifikasi Kasus Penggunaan Melihat Pemesanan..	40
Tabel 5.1 Hasil Uji Coba Jumlah Iterasi .....	98
Tabel 5.2 Hasil Pengujian Hidden Layer .....	99
Tabel 5.3 Hasil Pengujian Learning Rate.....	100
Tabel 5.4 Skenario Uji Coba Melakukan Pencarian Indeks....	102
Tabel 5.5 Skenario Uji Coba Melakukan Pemesanan Indeks..	103
Tabel 5.6 Skenario Uji Coba Melakukan Pembatalan Pemesanan Indeks .....	103
Tabel 5.7 Skenario Melakukan Konfirmasi Pembayaran Indeks .....	104
Tabel 5.8 Skenario Uji Coba Mendapatkan E-Ticket .....	105
Tabel 5.9 Skenario Uji Coba Melihat Kamar Indeks .....	106
Tabel 5.10 Skenario Uji Coba Melihat Indeks .....	107
Tabel 5.11 Skenario Uji Coba Melihat Transaksi .....	108

Tabel 5.12 Skenario Uji Coba Melihat Laporan Keuangan .....	110
Tabel 5.13 Skenario Uji Coba Melihat Pemesanan .....	110
Tabel 5.14 Daftar Partisipan .....	113
Tabel 5.15 Penentuan Waktu pada Uji Coba .....	113
Tabel 5.16 Harga Indekos yang Dipilih Partisipan pada Uji Coba .....	114
Tabel 5.17 Hasil Uji Coba Tingkat Kepuasan Terhadap Harga Baru .....	116
Tabel 5.18 Hasil Uji Coba Pengukuran Keuntungan atau Kerugian .....	116
Tabel 5.19 Hasil Evaluasi Uji Coba Back Propagation .....	118
Tabel 5.20 Evaluasi Uji Coba Fungsional Pencari Indekos .....	118
Tabel 5.21 Evaluasi Uji Coba Fungsionalitas .....	119

## **DAFTAR KODE SUMBER**

Kode Sumber 4.1 Training Data.....	81
Kode Sumber 4.2 Testing Data .....	82
Kode Sumber 4.3 Fungsi Proses Pencarian.....	84
Kode Sumber 4.4 Fungsi Pemesanan Kamar Indekos.....	86
Kode Sumber 4.5 Fungsi Pembatalan Pemesanan .....	87
Kode Sumber 4.6 Fungsi Pembayaran .....	87
Kode Sumber 4.7 Fungsi Mendapatkan E-Ticket .....	88
Kode Sumber 4.8 Fungsi Membuat E-Ticket.....	91
Kode Sumber 4.9 Fungsi Lihat Kamar Indekos .....	92
Kode Sumber 4.10 Fungsi Melihat Indekos .....	93
Kode Sumber 4.11 Fungsi Melihat Transaksi .....	93
Kode Sumber 4.12 Fungsi Verifikasi Pembayaran .....	94
Kode Sumber 4.13 Fungsi Melihat Laporan Keuangan .....	94
Kode Sumber 4.14 Fungsi Melihat Pemesanan.....	95

*[Halaman sengaja dikosongkan]*



# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Seiring dengan perkembangan teknologi informasi di Indonesia sektor bisnis pun semakin berkembang dan selalu ada perubahan yang nyata setiap harinya. Perubahan yang terjadi merupakan strategi pengelolaan bisnis yang sudah menggunakan teknologi untuk media bisnis. Hal ini dilakukan untuk mendapatkan hasil yang lebih baik. Apalagi dengan semakin ketatnya persaingan di dunia bisnis, membuat para pelaku bisnis harus mempunyai inovasi yang baru dan meningkatkan mutu bisnisnya. Pengusaha menyadari akan pentingnya sebuah informasi yang dapat membantu dalam mengambil suatu keputusan yang penting untuk perkembangan bisnis. Dalam perkembangannya, bisnis *e-commerce* muncul untuk mengatasi segala keterbatasan yang dapat terjadi di bisnis konvensional. Bisnis *e-commerce* baik untuk ditekuni oleh pengusaha sebab *e-commerce* membuat bisnis dapat lebih efisien dan efektif dalam meningkatkan keuntungan.

Peningkatan jumlah mahasiswa yang merantau membuat permintaan indekos meningkat di daerah sekitar kampus. Ini membuat bisnis indekos dapat dipertimbangkan sebagai bisnis yang menguntungkan. Apalagi indekos merupakan kebutuhan primer bagi mahasiswa perantauan, sehingga bisnis indekos tidak akan sepi peminat dan permintaan indekos akan selalu meningkat. Ketidak seimbangannya ketersediaan indekos dengan permintaan indekos membuat persaingan antar mahasiswa untuk mendapatkan indekos sangat ketat. Hal ini menyebabkan para mahasiswa pencari indekos yang tersedia di penghujung waktu sangat sulit.

Selain kesusahan mencari lokasi, harga menjadi salah satu faktor pendukung mahasiswa untuk memilih indekos. Saat ini kebanyakan indekos menawarkan harga yang flat tidak seperti hotel yang mempunyai harga fluktuatif di waktu tertentu. Jika

indekos mengimplementasikan harga seperti harga hotel dapat menarik mahasiswa.

Dalam mengatasi hal-hal tersebut penulis mengajukan beberapa penyelesaian melalui Tugas Akhir ini dengan membangun aplikasi *e-commerce* Carikos pada modul penetapan harga menggunakan back propagation.

## **1.2 Rumusan Masalah**

Tugas Akhir ini mengangkat beberapa rumusan masalah sebagai berikut:

1. Bagaimana implementasi metode back propagation pada penentuan harga indekos?
2. Bagaimana menghasilkan eror yang minimal pada metode back propagation?
3. Bagaimana membuat harga indekos yang fluktuatif dari hasil back propagation?

## **1.3 Batasan Permasalahan**

Permasalahan yang dibahas pada Tugas Akhir ini memiliki batasan sebagai berikut:

1. Data masukan back propagation berupa permintaan dan nilai bobot indekos.
2. Data masukan permintaan dan nilai bobot indekos dinormalisasi terlebih dahulu.
3. Aplikasi CariKos berbasis web dengan kerangka kerja CodeIgniter 3.1.2.
4. Jangkauan aplikasi hanya sekitar Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

## **1.4 Tujuan**

Tujuan dari Tugas Akhir ini adalah sebagai berikut:

1. Mengimplementasikan metode back propagation pada penentuan harga indekos.
2. Menghitung dengan metode back propagation hingga mendapatkan eror yang minimal.
3. Menetapkan harga indekos yang fluktuatif menggunakan nilai output dari back propagation dengan eror yang seminimal mungkin

## **1.5 Manfaat**

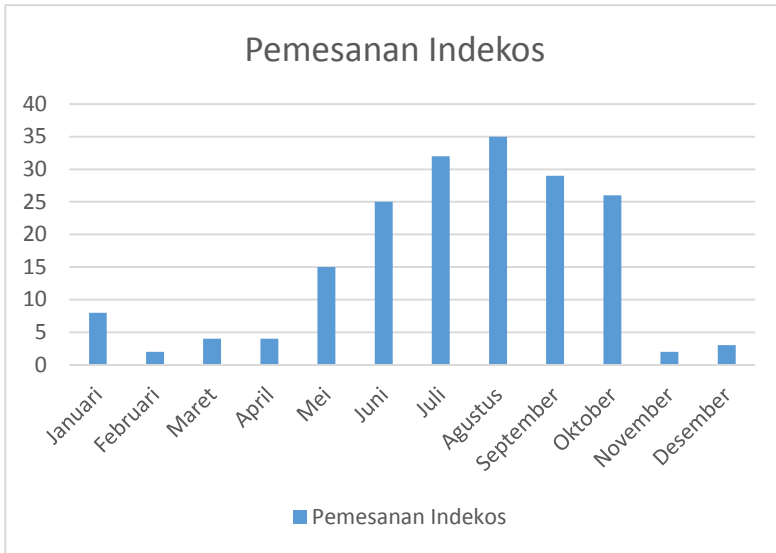
Manfaat yang diperoleh dari pembuatan Tugas Akhir ini adalah dapat memudahkan mahasiswa perantau untuk mendapatkan indekos yang berkualitas dan dekat dengan tempat perkuliahan dengan harga yang menarik.

## **1.6 Metodologi**

Pembuatan Tugas Akhir ini dilakukan dengan menggunakan metodologi sebagai berikut:

### **1.6.1 Penyebaran Kuesioner**

Penyebaran kuesioner ini digunakan penulis sebagai tolak ukur, pada bulan apa mahasiswa melakukan pencari indekos. Ini menjadi pertimbangan untuk menentukan periode permintaan agar lebih sesuai dengan kenyataan. Kuesioner ini dilakukan ke 50 responden. Dari 50 responden, sebanyak 35 responden menyatakan dalam mencari indekos pada bulan Agustus yang merupakan bulan pertama di semester baru. Sedangkan bulan Februari dan November hanya dipilih oleh responden. Hasil dari kuesioner ini dapat dilihat di Gambar 1.1 dan pertanyaannya dapat dilihat di Tabel A.1.



**Gambar 1.1 Diagram Permintaan Indekos**

### 1.6.2 Studi Literatur

Tugas Akhir ini menggunakan literatur *paper* yang berasal dari jurnal internasional dan jurnal ilmiah untuk mencari informasi yang dapat dijadikan referensi dalam pengerjaan Tugas Akhir ini. Selain itu juga digunakan sejumlah referensi buku dan literatur lain yang berhubungan dengan penerapan metode back propagation.

### 1.6.3 Analisis dan Desain Perangkat Lunak

Aplikasi ini merupakan aplikasi yang mempermudah pencarian indekos. Terdapat dua aktor yaitu, pencari indekos dan admin dari aplikasi. Fitur dari aplikasi ini antara lain pengguna dapat mencari indekos yang sesuai dengan keinginan, dapat melakukan pemesanan indekos, dan dapat melakukan pembayaran indekos.

### 1.6.4 Implementasi Perangkat Lunak

Implementasi merupakan tahap untuk membangun metode-metode yang sudah diajukan pada proposal Tugas Akhir. Untuk membangun algoritma yang telah dirancang sebelumnya, maka dilakukan implementasi dengan menggunakan suatu perangkat lunak. Pada tugas akhir ini, penulis menggunakan bahasa PHP, HTML, CSS, JavaScript dengan bantuan perangkat lunak Sublime Text. *Framework* yang digunakan adalah Code Igniter. Perangkat lunak untuk membuat database menggunakan phpMyadmin dan dengan server lokal XAMPP.

### 1.6.5 Pengujian dan Evaluasi

Pada tahap ini metode dari perangkat lunak yang telah disusun dan diimplementasikan akan dilakukan uji coba. Uji coba akan dilakukan oleh lima partisipan untuk mengukur keberhasilan dari fungsionalitas aplikasi, melihat tingkat kepuasan, dan melihat keuntungan/kerugian yang didapat dalam satu periode jika dilakukan pemesanan. Hal ini dilakukan untuk mendapatkan hasil keuntungan atau kerugian dari pemesanan yang dilakukan. Selain itu dilakukan juga uji coba terhadap back propagation untuk mendapatkan *weight*, *learning rate*, dan iterasi yang optimal, sehingga mendapatkan nilai MSE yang kecil. Jika mendapatkan keuntungan maka metode back propagation cocok untuk penentuan harga dan sebaliknya jika mendapatkan kerugian maka metode back propagation kurang cocok untuk penentuan harga.

## 1.7 Sistematika Penulisan

Buku Tugas Akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan Tugas Akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku Tugas Akhir terdiri atas beberapa bagian seperti berikut ini.

**Bab I        Pendahuluan**

Bab ini berisi latar belakang masalah, tujuan dan manfaat pembuatan Tugas Akhir, permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penyusunan Tugas Akhir.

**Bab II        Dasar Teori**

Bab ini membahas beberapa teori penunjang yang berhubungan dengan pokok pembahasan dan mendasari pembuatan Tugas Akhir ini.

**Bab III       Analisis dan Perancangan Sistem**

Bab ini membahas mengenai perancangan perangkat lunak. Perancangan perangkat lunak meliputi perancangan data, arsitektur, proses dan perancangan antarmuka aplikasi.

**Bab IV       Implementasi**

Bab ini berisi implementasi dari perancangan dan implementasi fitur-fitur penunjang aplikasi.

**Bab V        Pengujian dan Evaluasi**

Bab ini membahas pengujian dengan metode kotak hitam (*black box testing*) untuk mengetahui aspek nilai fungsionalitas dari perangkat lunak dan nilai kegunaan yang dibuat dengan juga memperhatikan ketertarikan pada calon partisipan untuk menggunakan aplikasi ini.

**Bab VI       Kesimpulan dan Saran**

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan. Bab ini membahas saran-saran untuk pengembangan sistem lebih lanjut.

**Daftar Pustaka**

Merupakan daftar referensi yang digunakan untuk mengembangkan Tugas Akhir.

**Lampiran**

Merupakan lampiran.

## **BAB II**

### **TINJAUAN PUSTAKA**

Bab ini berisi pembahasan mengenai teori-teori dasar yang digunakan dalam Tugas Akhir. Teori-teori tersebut diantaranya adalah back propagation, *e-commerce*, dan beberapa teori lain yang mendukung pembuatan Tugas Akhir.

#### **2.1 Aplikasi Sejenis**

Aplikasi CariKos yang akan dibangun merupakan aplikasi pencarian indekos yang akan digunakan oleh para pencari indekos. Saat ini ada aplikasi sejenis yang terlebih dahulu ada yaitu aplikasi mamikos.com. Aplikasi mamikos.com hanya menyajikan informasi kamar indekos, beserta fasilitas indekos, harga indekos, dan foto kamar indekos sesuai dengan kondisi sebenarnya. Sehingga ketika pencari indekos ingin memesan indekos, pencari indekos harus langsung menuju lokasi indekos yang dipesannya.

Selain aplikasi mamikos.com, aplikasi CariKos juga hampir mirip dengan aplikasi pemesanan hotel. Kesamaan antara keduanya adalah dalam penentuan harga yang fluktuatif sesuai dengan permintaan pasar. Tetapi keduanya memiliki perbedaan yaitu, aplikasi CariKos untuk melakukan pencarian dan pemesanan indekos sedangkan aplikasi pemesanan hotel melakukan pemesanan hotel.

Melihat kekurangan dari aplikasi mamikos.com tersebut, lahirlah ide untuk membuat aplikasi sejenis mamikos.com yaitu aplikasi CariKos. Aplikasi CariKos akan mempunyai fitur untuk melakukan pemesanan dan pembayaran indekos melalui aplikasi. Sehingga pencari indekos tidak harus datang langsung ke lokasi indekos untuk melakukan pemesanan indekos.

## 2.2 E - Commerce

*Electronic commerce (e-commerce)* adalah proses pembelian, penjualan atau pertukaran produk, jasa dan informasi melalui jaringan computer. Selain teknologi jaringan *www*, *e-commerce* juga memerlukan teknologi basis data atau pangkalan data (*database*), surat elektronik (*e-mail*), dan bentuk teknologi non computer yang lain seperti halnya sistem pengiriman barang dan alat pembayaran untuk *e-commerce* ini. (Siregar, 2010)

Dalam mengimplementasikan *e-commerce* tersedia suatu integrasi rantai nilai dari infrastrukturnya, yang terdiri dari tiga lapis. Pertama, infrastruktur sistem distribusi (*flow of good*). Kedua, infrastruktur pembayaran (*flow of money*). Dan ketiga, infrastruktur sistem informasi (*flow of information*) [1]. Ada tiga faktor yang dapat dicermati jika ingin membangun toko *e-commerce* yaitu : *variability*, *visibility*, dan *velocity*.

Di Indonesia *e-commerce* mulai diminati oleh para pengusaha. Dari data yang dirilis biro riset Frost & Sullivan, bersama China, Indonesia menjadi negara dengan pertumbuhan pasar *e-Commerce* terbesar di dunia dengan rata-rata pertumbuhan 17 persen setiap tahun. Menurut CEO DOKU Thong Sennelius, pesatnya pertumbuhan pasar *e-commerce* di Indonesia didukung oleh antusiasme terhadap platform media social dikarenakan sistem pemasaran melalui media sosial jauh lebih bertenaga dibandingkan sistem pemasaran produk konvensional, atau hanya memajangkannya di website [2]. Dari data di atas dapat disimpulkan bahwa pasar *e-commerce* sangat menjanjikan untuk pelaku usaha membuka bisnis baru. Kemudahan untuk membuka toko *online* menjadi kelebihan dari pasar *e-commerce*. Jenis bisnis yang bisa dibuka pun beragam, mulai dari menjual barang/produk ataupun menawarkan jasa untuk konsumen. Penentuan membuka bisnis baru dapat mempertimbangkan permintaan pasar akan suatu barang atau jasa. Jika permintaan suatu barang atau jasa tinggi, maka ada peluang untuk membuka bisnis tersebut.



*E-commerce* dapat dibagi menjadi beberapa jenis yang memiliki karakteristik berbeda-beda :

### **2.2.1 Business to Business (B2B)**

*Business to Business* memiliki karakteristik:

- *Trading partner* yang sudah ada dan mempunyai hubungan lama sehingga pertukaran informasi dapat disusun sesuai dengan kebutuhan dan kepercayaan (*trust*).
- Pertukaran data berlangsung secara berulang-ulang dan berkala dengan format yang sudah disepakati.
- Pengiriman data tidak perlu menunggu permintaan dari *partner*-nya.
- Model yang digunakan adalah *peer-to-peer*, dimana *processing intelligence* dapat didistribusikan kepada kedua pelaku usaha.

Umumnya *e-commerce* jenis ini menggunakan mekanisme *Electronic Data Interchange* (EDI). EDI adalah proses transfer data yang terstruktur dalam format standar yang disetujui dari satu komputer ke komputer lainnya dalam bentuk elektronik. Contoh *e-commerce* jenis ini di Indonesia adalah [www.bizzy.co.id](http://www.bizzy.co.id) dan [www.ralali.com](http://www.ralali.com).

### **2.2.2 Business to Consumer (B2C)**

*Business to Consumer* memiliki karakteristik sebagai berikut:

- Terbuka untuk umum.
- Pelayanan yang diberikan bersifat umum.
- Layanan yang diberikan berdasarkan permintaan konsumen.
- Menggunakan pendekatan sistem *client/server* dimana *client* (konsumen) menggunakan sistem berbasis web dan *processing (business procedure)* diletakkan di sisi server.

Pada *e-commerce* jenis ini berkembang sangat pesat ditandai dengan lahirnya berbagai toko *online* di Indonesia. Toko

*online* menggunakan web sebagai toko *virtual* untuk menjajakan produk dan layanan. Contoh dari *e-commerce* jenis ini di Indonesia adalah [www.traveloka.com](http://www.traveloka.com) dan [www.tiket.com](http://www.tiket.com).

### **2.2.3 Consumer to Consumer (C2C)**

*E-commerce* jenis ini meliputi semua transaksi elektronik barang atau jasa antar konsumen. Kebanyakan di *e-commerce* jenis ini menyediakan tempat bagi penjual untuk menjualkan langsung barangnya. *E-commerce* ini pun yang paling terkenal di Indonesia. Contoh dari *e-commerce* jenis ini di Indonesia adalah [www.bukalapak.com](http://www.bukalapak.com) dan [www.tokopedia.com](http://www.tokopedia.com).

### **2.2.4 Consumer to Business (C2B)**

*E-commerce* jenis ini merupakan jenis *e-commerce* yang menyediakan tempat bagi individu untuk menawarkan jasa atau produk mereka kepada perusahaan yang sedang mencari jasa atau produk tersebut. Contoh dari *e-commerce* jenis ini di Indonesia adalah [www.istockphoto.com](http://www.istockphoto.com).

### **2.2.5 Business to Administration (B2A)**

*E-commerce* jenis ini mencakup semua transaksi *online* yang dilakukan antara perusahaan dan administrasi publik. Contoh *e-commerce* jenis ini di Indonesia adalah [www.pajak.go.id](http://www.pajak.go.id) dan [www.bpjs-online.com](http://www.bpjs-online.com).

### **2.2.6 Consumen to Administration (C2A)**

*E-commerce* jenis ini hampir mirip dengan *business to Administration* (B2A) tetapi yang terlibat bukan antara perusahaan dan administrasi publik melainkan antara individu dan administrasi

publik. Contoh dari *e-commerce* jenis ini di Indonesia adalah [www.pajak.go.id](http://www.pajak.go.id).

## 2.3 Teori Permintaan

Mempertimbangkan permintaan pasar terhadap barang atau jasa diperlukan ketika kita membuka bisnis baru. Penentuan harga pun menjadi persoalan yang rumit. Jika harga semakin murah maka permintaan akan semakin banyak, tetapi penawaran semakin menurun. Ini terjadi karena ingin mencari keuntungan sebesar-besarnya. Apabila harga terlalu tinggi maka pembeli mungkin akan membeli dalam jumlah sedikit, namun berbeda dengan penjual yang akan memperbanyak barang yang dijual atau diproduksi agar mendapatkan keuntungan yang besar. Harga yang tinggi juga bisa menyebabkan konsumen akan mencari produk lain sebagai pengganti barang yang harganya mahal [3].

Dalam ekonomi terdapat satu titik pertemuan antara permintaan (*demand*) dan penawaran (*supply*) dalam satuan harga dan jumlah barang. Hukum permintaan menjelaskan hubungan antara perubahan harga suatu barang dengan perubahan barang yang diminta. Hukum permintaan menyatakan bahwa antara harga dan jumlah barang yang diminta berbanding terbalik. Artinya bila harga naik akan menyebabkan jumlah barang yang diminta akan turun dan sebaliknya, dengan syarat *ceteris paribus* [3].

Permintaan suatu barang bisa dikatakan elastis atau inelastis didasari atau ditentukan oleh berbagai faktor yaitu [3]:

### 1. Barang Mewah dan Barang Kebutuhan

Permintaan barang-barang kebutuhan umumnya inelastis, sedangkan permintaan barang-barang mewah umumnya elastis. Karena walaupun harga barang kebutuhan mengalami peningkatan atau penurunan, jumlah yang diminta akan tetap sama atau hanya mengalami penurunan sedikit. Berbeda dengan barang mewah jika mengalami peningkatan harga, jumlah yang diminta tidak ada. Tapi jika mengalami penurunan harga jumlah yang diminta akan meningkat.

## 2. Ketersediaan barang Substitusi

Apabila suatu barang memiliki barang pengganti atau barang substitusi akan mempunyai elastisitas yang elastis, sedangkan barang tidak memiliki pengganti atau substitusi mempunyai elastisitas yang inelastis. Sebab ketika barang itu mengalami peningkatan harga maka permintaan akan barang tersebut akan mengalami penurunan dikarenakan tersedianya barang substitusi yang harganya lebih rendah dari barang tersebut. Berbeda dengan barang yang tidak mempunyai barang substitusi akan mengalami penurunan permintaan yang sedikit.

## 3. Definisi Pasar

Semakin luas ruang lingkup pasarnya maka semakin inelastis barang tersebut karena tidak ada barang substitusinya. Sebaliknya, semakin sempit ruang lingkup pasarnya maka semakin elastis barang tersebut.

## 4. Rentang Waktu

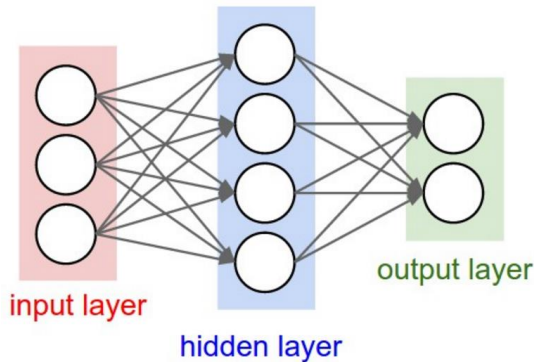
Apabila rentang waktu perubahan harga suatu barang lebih lama atau jangka panjang, permintaan barang tersebut akan elastis. Karena orang-orang (konsumen) mampu untuk mencari dan mensubstitusi barang tersebut dan biasa tidak menggunakan barang tersebut lagi. Namun, untuk jangka waktu yang pendek akan mengalami inelastis karena tidak adanya kesempatan bagi konsumen untuk mensubstitusi barang tersebut.

Dengan teori permintaan ini menjadi landasan untuk menganalisis harga yang wajar untuk indeks dengan mempertimbangkan permintaan akan kebutuhan indeks. Untuk menentukan sebuah harga maka dibutuhkan algoritma yang dapat memprediksi harga dengan mempertimbangkan beberapa pertimbangan.

## 2.4 Back Propagation Neural Network

Back Propagation Neural Network mempunyai kemampuan yang kuat dari interpolasi non-linear. Metode ini secara luas digunakan dalam aplikasi teknik untuk prediksi dan

optimalisasi. Back Propagation Neural Network terdiri dari *input layer*, beberapa *hidden layer* dan *output layer* [4]. Pada Gambar 2.1 dapat dilihat arsitektur back propagation.



Gambar 2.1 Arsitektur *Back Propagation*

*Input layer* terdiri dari nilai bias, nilai permintaan, dan nilai bobot indekos. Nilai bias merupakan nilai yang konstan dan mempunyai nilai 1. Nilai bias digunakan untuk membuat nilai *output* yang diinginkan menjadi lebih optimal. Dua *input* yaitu nilai permintaan dan nilai skor dari indekos. Nilai permintaan didapatkan dari hasil kuesioner yang telah dilakukan sebelumnya. Nilai permintaan dapat dilihat pada Gambar 1.1. Nilai bobot indekos merupakan nilai yang dipunya dari sebuah indekos berdasarkan kriteria. Kriterianya adalah dekat dengan supermarket atau minimarket, dekat dengan tempat ibadah, luas parkir yang dimiliki indekos, terdapatnya penjaga indekos, ketersediaan fasilitas yang ditawarkan, tidak berada di daerah banjir, tidak berada di daerah ramai, dan dekat dengan lokasi jurusan.

Terdapat 3 fase dalam pelatihan back propagation, yaitu fase maju (*feed forward*), fase mundur (*back propagation*), dan fase modifikasi bobot. Rumus untuk perhitungan bobot :

$$w_{lji}(k+1) = w_{lji}(k) - \mu \frac{\partial MSE(w)}{\partial w_{lji}}$$

Dimana  $w_{lji}$  adalah bobot antara neuron  $i$  pada layer  $l-1$  dan neuron  $j$  pada layer  $l$ ,  $\mu$  adalah bilangan real positif, learning rate yang berfungsi untuk mengontrol step pembelajaran algoritma dan biasanya diset kecil. Dalam fase back propagation, tiap-tiap unit output menerima target pola yang berhubungan dengan pola input untuk dihitung nilai kesalahan. Kesalahan tersebut akan dipropagasikan mundur. Sedangkan fase modifikasi bobot bertujuan untuk menurunkan kesalahan yang terjadi. Ketiga fase tersebut diulang secara terus menerus hingga kondisi penghentian dipenuhi.

Pada penelitian penentuan upah minimum kota berdasarkan tingkat inflasi menggunakan back propagation neural network, didapatkan nilai MSE sebesar 0.0728 [4]. Oleh sebab itu nilai MSE tersebut menjadi landasan eror minimal pada aplikasi CariKos. Maka dari itu, untuk menentukan harga indeks dapat menggunakan back propagation neural network. Back propagation melakukan pemodelan terhadap nilai permintaan dan nilai bobot indeks untuk menentukan harga baru.

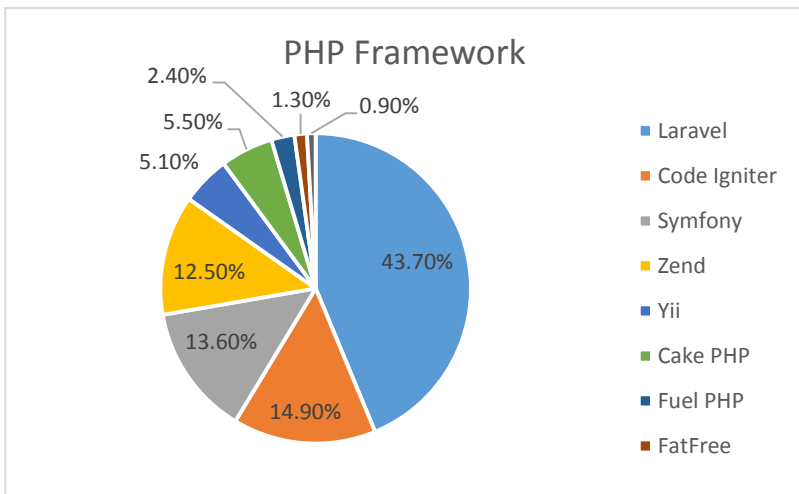
## 2.5 *CodeIgniter*

CodeIgniter adalah aplikasi *open source* yang merupakan *framework* dari bahasa pemrograman PHP dengan model MVC (*Model, View, Controller*). *Framework* secara sederhana dapat diartikan sebagai kumpulan dari fungsi – fungsi / prosedur – prosedur dan *class – class* yang sudah siap digunakan untuk mempermudah dan mempercepat pekerjaan programmer.

MVC merupakan suatu konsep yang cukup populer dalam pembangunan web yang memisahkan pengembangan aplikasi berdasarkan komponen utama seperti memanipulasi data, *user interface*, dan bagian yang menjadi kontrol aplikasi. Terdapat tiga jenis komponen yang membangun MVC yaitu :

1. *View* merupakan bagian yang menangani *presentation logic* yang mempunyai file bertipe HTML, yang diatur oleh *controller*. *View* berfungsi untuk menerima dan merepresentasikan data kepada *user*. Bagian ini tidak memiliki akses langsung terhadap bagian model.
2. *Controller* merupakan bagian yang mengatur hubungan antara bagian *view* dan *model*, *controller* berfungsi untuk menerima *request* dan data dari *user* kemudian menentukan proses yang akan dilakukan aplikasi.
3. *Model* berhubungan langsung dengan *database* untuk memanipulasi data (*insert*, *delete*, *update*), menangani validasi dari bagian *controller*, namun tidak dapat berhubungan langsung ke bagian *view*.

Selain CodeIgniter masih banyak *framework* yang tersedia seperti laravel, Yii, Symfony, Zend, Cake PHP, Fuel PHP, FatFree, Aura. Persentase penggunaan PHP *framework* dapat dilihat di Gambar 2.3. Di Gambar 2.3 dapat dilihat bahwa tiga PHP *framework* yang paling banyak digunakan adalah laravel, CodeIgniter, dan Zend [5].



**Gambar 2.2 Diagram Presentase Penggunaan PHP Framework**

## 2.6 Sublime Text

Sublime Text merupakan editor source code lintas platform dengan Python Application Programming Interface (API). Sublime Text mendukung berbagai macam bahasa pemrograman. Fitur dari Sublime Text [6]:

- *Goto Anything* merupakan navigasi cepat ke file, simbol atau baris.
- *Command Palette* merupakan pencocokan cepat aktif untuk *keyboard* untuk menjalankan perintah.
- *Simultaneous Editing* merupakan perubahan yang dilakukan bersamaan terhadap ke beberapa daerah yang dipilih.
- *Python-based plugin API*
- *Project-specific preferences*

## 2.7 Hypertext Preprocessor (PHP)

PHP adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak digunakan untuk memrogram web dinamis. PHP dapat digunakan untuk membangun sebuah CMS. Kelebihan PHP dari Bahasa pemrograman web yang lain [7]:

- Bahasa pemrograman PHP adalah sebuah bahasa skrip yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- Web Server yang mendukung PHP mudah didapatkan seperti apache, IIS, Lighttpd, dan Xitami dengan konfigurasi yang relative murah.
- Dalam sisi pemahaman, PHP adalah bahasa skrip yang paling mudah karena memiliki referensi yang banyak.
- PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan perintah-perintah sistem.
- PHP memiliki delapan tipe data, yaitu *Boolean*, *Integer*, *Float/Double*, *String*, *Array*, *Object*, *Resource*, *NULL*.



## 2.8 TCPDF

TCPDF adalah kelas PHP untuk menghasilkan PDF tanpa memerlukan pustaka eksternal tambahan. Fitur utama pada TCPDF:

- Tidak membutuhkan *library* khusus.
- Dapat menggunakan format halaman standar, format halaman yang dikostumisasi, kostumisasi margin dan ukuran satuan.
- UTF-8 Unicode dan Bahasa dari kiri ke kanan.
- TrueTypeUnicode, OpenTypeUnicode, TrueType, OpenType, Type1 and CID-0 fonts.
- Memiliki metode untuk mempublikasikan beberapa kode XHTML + CSS, JavaScript dan Form.
- Anotasi PDF, termasuk tautan, lampiran teks, dan file.
- Mendukung JPEG, PNG, dan SVG.

Oleh karena itu, TCPDF dapat digunakan untuk menghasilkan *e-booking* dari pemesanan kamar indekos.

*[Halaman ini sengaja dikosongkan]*

## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

Bab ini membahas tahap analisis dan perancangan sistem yang akan dibangun. Analisis membahas semua persiapan yang akan menjadi pokok pikiran pembuatan aplikasi ini. Mulai dari masalah yang melatarbelakangi, hingga analisis gambaran awal sistem yang akan dibuat. Perancangan sistem membahas hal-hal yang berkaitan dengan pondasi atau dasar pembuatan aplikasi, yang meliputi perancangan basis data, tampilan antar muka halaman aplikasi, hingga perancangan alur proses yang akan diimplementasikan di dalam aplikasi.

#### **3.1 Analisis**

Tahap analisis meliputi analisis masalah, analisis kebutuhan, deskripsi umum sistem, dan kasus penggunaan sistem yang dibuat.

##### **3.1.1 Analisis Permasalahan**

Kesulitan pencarian indekos merupakan permasalahan dari para pencari indekos. Pencari indekos diharuskan mencari langsung lokasi indekos yang tersedia. Sedangkan tidak semua pencari indekos mengetahui daerah atau lokasi di kota tujuan.

Walaupun sudah tersedia aplikasi pencarian indekos saat ini. Aplikasi tersebut hanya memberikan informasi ketersediaan indekos tanpa menyediakan fitur pemesanan pada aplikasinya. Sehingga para pencari indekos harus datang ke tempat indekos yang dipilih untuk melakukan pemesanan.

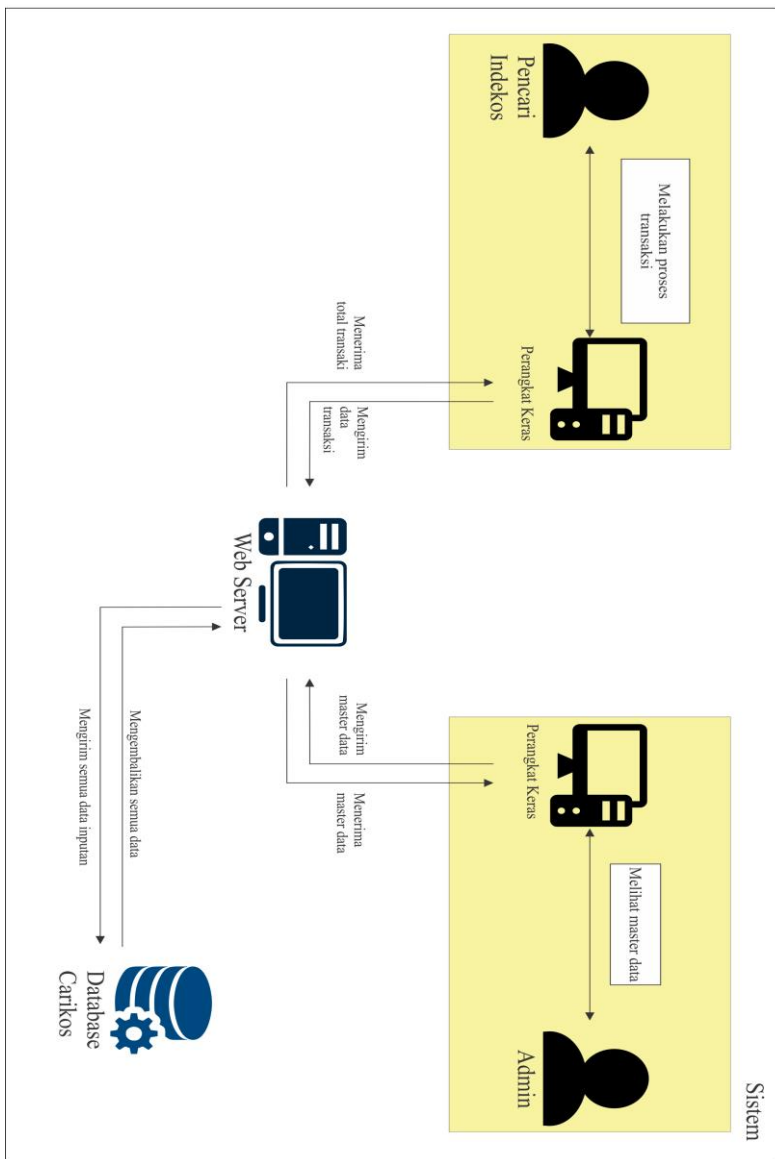
Dari dua permasalahan di atas, terlahirlah ide untuk membuat aplikasi CariKos yang dapat melakukan pencarian indekos dan pemesanan indekos. Sehingga mempermudah para pencari indekos untuk melakukan pencarian indekos dan melakukan pemesanan indekos tanpa harus datang langsung ke indekos yang diinginkan.

### 3.1.2 Deskripsi Umum Sistem

CariKos merupakan aplikasi berbasis web pencari indekos. Di aplikasi ini juga dapat melayani pemesanan indekos sehingga pencari tidak perlu pergi ke tempat indekos yang diinginkan. Selain itu aplikasi ini melayani juga transaksi untuk membayar pemesanan indekos. Pencarian indekos dapat disesuaikan dengan keinginan dari pencari indekos. Pencari indekos dapat memilih rentang harga sewa indekos, tipe indekos, fasilitas indekos yang diinginkan, fasilitas kamar yang diperlukan, dekat dengan jurusan yang diinginkan. Pencarian yang dapat disesuaikan ini digunakan agar indekos yang ingin dicari pencari indekos dapat sesuai kriteria yang diinginkan.

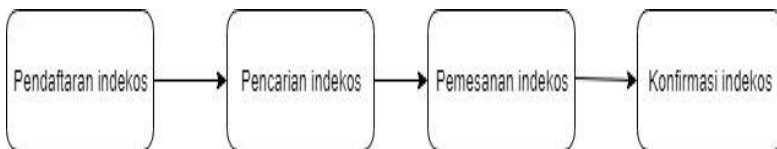
Harga yang ditampilkan di aplikasi ini tidak selalu sama disetiap bulannya. Harga naik-turun sesuai dengan permintaan di bulan itu sesuai dengan hasil kuesioner dan analisis permintaan indekos. Selain permintaan, kualitas indekos juga menjadi dasar penentuan harga. Kualitas indekos dapat dilihat dari seberapa besar indekos mendapatkan skor.

Arsitektur pada aplikasi indekos dapat dilihat pada Gambar 3.1. Pada Gambar 3.1 dapat dilihat hubungannya antara pengguna, perangkat keras dan perangkat lunak. Proses diawali oleh para pemilik indekos yang mendaftarkan indekosnya pada aplikasi CariKos melalui *browser* pada komputer. Lalu informasi indekos yang didaftarkan tadi dikirim ke *web server* untuk disimpan ke dalam *database* CariKos. Proses yang lainnya adalah pencari indekos melakukan pencarian indekos. Informasi indekos yang dicari oleh pencari indekos dikirim ke *web server* untuk dilihat ketersediaan indekos dalam *database* Carikos.



Gambar 3.1 Arsitektur Aplikasi CariKos

Pada aplikasi CariKos terdapat empat proses bisnis proses utama yaitu, pendaftaran indekos, pencarian indekos, pemesanan indekos, dan konfirmasi pembayaran. Gambar 3.2 merupakan bisnis proses dari aplikasi CariKos. Proses pertama, pemilik indekos dapat mendaftarkan indekos miliknya ke aplikasi CariKos. Di pendaftaran indekos, pemilik akan diminta memasukkan informasi indekos berupa nama indekos, alamat indekos, nomor telepon indekos, kamar indekos, fasilitas indekos dan kamar indekos, foto indekos dan kamar indekos, beserta harga yang ditawarkan untuk menyewa indekos perbulannya. Selanjutnya, pencari indekos dapat mencari indekos sesuai dengan kriteria yang diinginkan. Ketika setelah melakukan pencarian indekos, pencari indekos dapat langsung melakukan pemesanan indekos dan dapat melakukan transaksi pembayaran dengan melakukan *transfer* ke nomor rekening CariKos. Setelah melakukan *transfer*, pencari indekos melakukan konfirmasi pembayaran pada aplikasi. Setelah pihak CariKos sudah menerima pembayaran, maka pencari indekos dianggap sudah membayar dan mendapatkan *e-ticket* pemesanan indekos.



**Gambar 3.2 Proses Bisnis Aplikasi CariKos**

Aplikasi CariKos dikerjakan oleh tiga pengembang. Untuk Tugas Akhir ini lebih difokuskan kepada bisnis proses pencarian indekos, pemesanan indekos, dan konfirmasi pembayaran. Sehingga kedepannya tidak akan membahas proses bisnis pendaftaran indekos.

### 3.1.3 Analisis Kebutuhan Sistem

Kebutuhan utama dalam aplikasi ini adalah pencari indeks dapat melakukan pemesanan indeks, melakukan pembayaran dan mendapatkan *e-ticket* ketika sudah membayar. Selain itu, admin dapat melihat kamar indeks, indeks, pemilik indeks, histori transaksi dan juga laporan keuangan dari CariKos. Secara rinci, kebutuhan sistem dapat dilihat di Tabel 3.1.

**Tabel 3.1 Daftar Kebutuhan Sistem**

<b>Kode Kebutuhan</b>	<b>Kebutuhan Fungsional</b>	<b>Deskripsi</b>
F-0001	Melakukan pencarian indeks	Pencari indeks melakukan pencarian indeks yang sesuai keinginan
F-0002	Melakukan pemesanan indeks	Pencari indeks dapat melakukan pemesanan indeks yang sudah dipilih
F-0003	Melakukan pembatalan indeks	Pencari indeks dapat melakukan pembatalan pemesanan indeks
F-0004	Melakukan pembayaran Indeks	Pencari indeks dapat melakukan pembayaran terhadap pemesanan yang sudah dilakukan
F-0005	Mendapatkan <i>e-ticket</i>	Pencari indeks mendapatkan <i>e-ticket</i> bila sudah membayar pemesanan
F-0006	Melihat informasi indeks	Admin dapat melihat informasi indeks dan kamar indeks
F-0007	Melihat transaksi	Admin dapat melihat transaksi yang terjadi
F-0008	Verifikasi pembayaran	Admin dapat verifikasi pembayaran

F-0009	Melihat laporan keuangan	Admin dapat melihat laporan keuangan perbualan dari CariKos
F-0010	Melihat pemesanan	Admin dapat melihat pemesanan.

### 3.1.4 Analisis Aktor

Aktor adalah pihak-pihak, baik manusia maupun sistem yang terlibat dan berinteraksi langsung dengan sistem. Pada aplikasi CariKos ini memiliki dua aktor yaitu pencari indekos dan admin.

### 3.1.5 Kasus Penggunaan

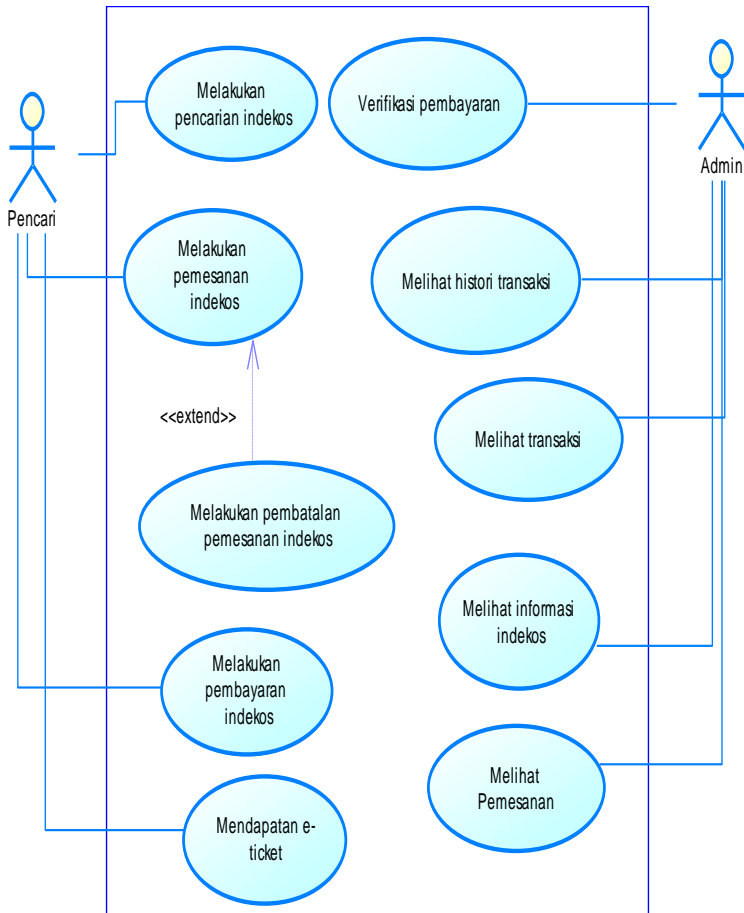
Mengacu pada spesifikasi kebutuhan fungsional yang telah dipaparkan, dibuat kasus penggunaan yang selanjutnya akan disimpulkan dalam deskripsi umum sistem, yang diharapkan dapat memenuhi kebutuhan fungsional, berdasar pada kasus penggunaan yang dibuat. Kasus penggunaan dijelaskan lebih lanjut pada Tabel 3.2 dan Gambar 3.3.

**Tabel 3.2 Daftar Kasus Penggunaan**

Kode Kasus Penggunaan	Nama	Aktor
UC-0001	Melakukan Pencari Indekos	Pencari Indekos
UC-0002	Melakukan Pemesanan Indekos	Pencari Indekos
UC-0003	Melakukan Pembatalan Pemesanan Indekos	Pencari Indekos
UC-0004	Melakukan Pembayaran Indekos	Pencari Indekos
UC-0005	Mendapatkan <i>e-ticket</i>	Pencari Indekos
UC-0006	Melihat informasi indekos	Admin
UC-0007	Melihat transaksi	Admin



UC-0008	Verifikasi pembayaran	Admin
UC-0009	Melihat laporan keuangan	Admin
UC-0010	Melihat pemesanan	Admin



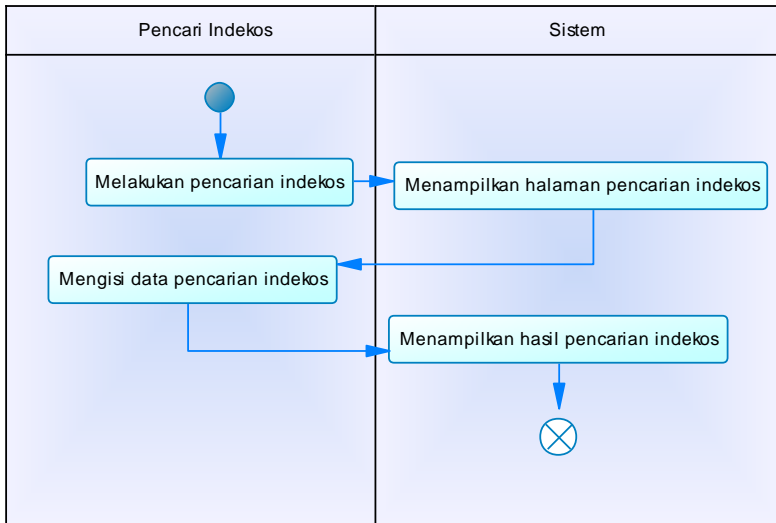
**Gambar 3.3. Diagram Kasus Penggunaan**

### 3.1.5.1 Melakukan Pencari Indekos (UC-0001)

Pada kasus penggunaan ini, pencari indekos dapat mencari indekos yang sesuai dengan keinginannya. Spesifikasi kasus penggunaan melakukan pencarian indekos dapat dilihat pada Tabel 3.3 dan diagram aktivitas pada Gambar 3.4.

**Tabel 3.3 Spesifikasi Kasus Penggunaan Melakukan Pencarian Indekos**

Kode	UC-0001
Nama	Melakukan Pencarian Indekos
Aktor	Pencari Indekos
Deskripsi	Pencari indekos mencari indekos sesuai dengan kebutuhannya
Tipe	Fungsional
Kondisi Awal	Pencari melakukan pencarian indekos
Kondisi Akhir	Sistem menampilkan hasil pencarian indekos
Alur Kejadian Normal	
Pencari Indekos	Sistem
1. Melakukan pencarian indekos.	2. Menampilkan halaman pencarian indekos.  4. Menampilkan hasil pencarian indekos.
3. Mengisi data pencarian indekos.	
Alur Kejadian Alternatif	
Pencari Indekos	Sistem
Tidak Ada	Tidak Ada



**Gambar 3.4. Diagram Aktivitas Melakukan Pencarian Indeks**

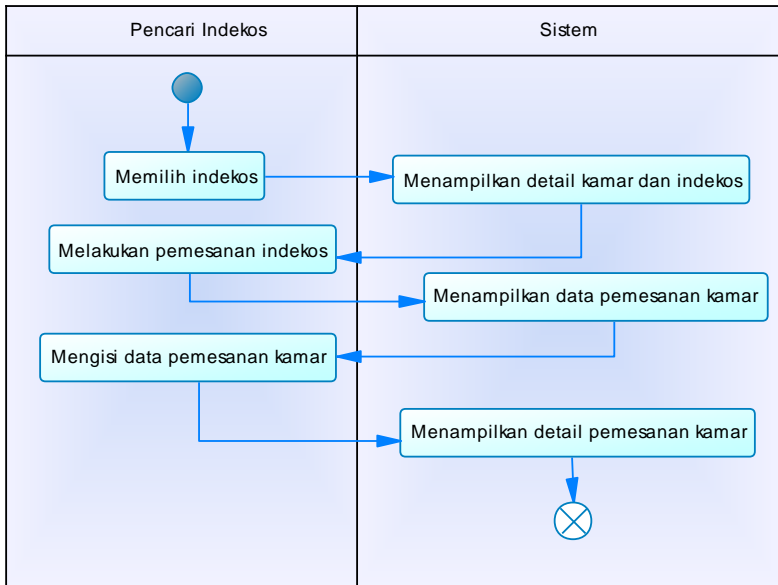
### 3.1.5.2 Melakukan Pemesanan Indeks (UC-0002)

Pada kasus penggunaan ini, pencari indeks dapat melakukan pemesanan indeks. Pemesanan indeks dapat dilakukan setelah melakukan pencarian indeks. Dalam melakukan pemesanan ini pencari indeks melakukan pengisian formulir pemesanan. Setelah melakukan pengisian formulir pemesanan maka pemesanan sudah dilakukan. Untuk lebih jelasnya spesifikasi kasus penggunaan melakukan pemesanan indeks dapat dilihat pada Tabel 3.4 dan diagram aktivitas pada Gambar 3.5.

**Tabel 3.4 Spesifikasi Kasus Penggunaan Melakukan Pemesanan Indeks**

Kode	UC-0002
Nama	Melakukan Pemesanan Indeks
Aktor	Pencari Indeks
Deskripsi	Pencari indeks dapat melakukan pemesanan indeks
Tipe	Fungsional

Kondisi Awal	Pencari indekos memilih indekos
Kondisi Akhir	Sistem menampilkan detail pemesanan kamar
Alur Kejadian Normal	
Pencari indekos	Sistem
1. Memilih indekos.  3. Pencari indekos melakukan pemesanan indekos.  5. Pencari indekos mengisi data pemesanan kamar.	2. Sistem menampilkan detail kamar dan indekos.  4. Menampilkan data pemesanan kamar.  6. Menampilkan detail pemesanan kamar.
Alur Kejadian Alternatif	
Pencari indekos	Sistem
Tidak ada.	Tidak ada.



**Gambar 3.5 Diagram Aktivitas Melakukan Pemesanan Indeks**

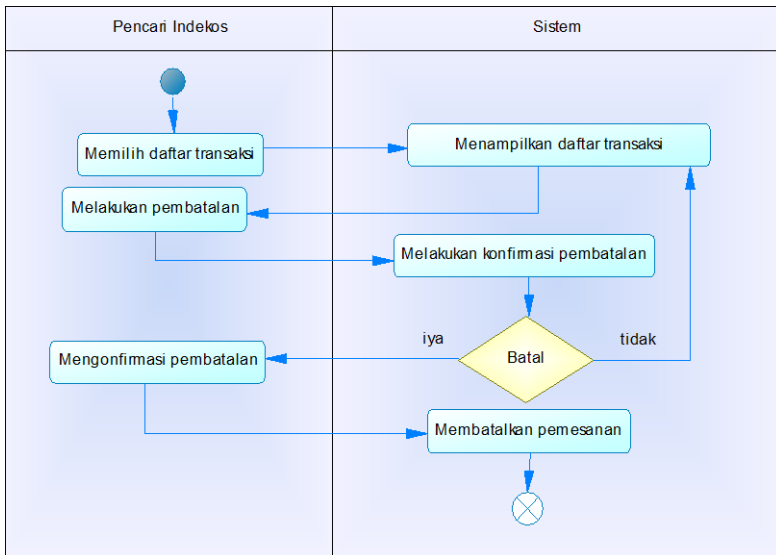
### 3.1.5.3 Melakukan Pembatalan Pemesanan Indeks (UC-0003)

Pada kasus penggunaan ini, bila pencari indeks sudah terlanjur melakukan pemesanan tetapi ternyata indeks yang dipilihnya tidak sesuai dengan keinginan ataupun mendapatkan indeks yang lebih baik dari yang sudah dipesan. Maka pencari indeks dapat melakukan pembatalan pemesanan indeks di dalam aplikasi CariKos. Pembatalan pemesanan indeks dilakukan dengan merubah status dalam *database* menjadi batal sehingga pemesanan dapat dibatalkan. Tetapi pembatalan dapat dilakukan sebelum pencari indeks melakukan pembayaran dan konfirmasi pembayaran. Jika pencari indeks sudah melakukan pembayaran dan konfirmasi pembayaran maka pencari indeks tidak dapat melakukan pemesanan indeks. Spesifikasi kasus penggunaan

melakukan pemesanan indeks dapat dilihat pada Tabel 3.5 dan diagram aktivitas pada Gambar 3.6.

**Tabel 3.5 Spesifikasi Kasus Penggunaan Melakukan Pembatalan Pemesanan Indeks**

Kode	UC-0003
Nama	Melakukan Pembatalan Pemesanan Indeks
Aktor	Pencari Indeks
Deskripsi	Pencari indeks membatalkan pemesanan indeks jika pencari indeks belum melakukan pembayaran dan konfirmasi pembayaran
Tipe	Fungsional
Kondisi Awal	Pencari indeks memilih daftar transaksi
Kondisi Akhir	Sistem membatalkan pemesanan
Alur Kejadian Normal	
Pencari indeks	Sistem
1. Memilih daftar transaksi.	2. Menampilkan daftar transaksi.
3. Melakukan pembatalan pemesanan.	4. Melakukan konfirmasi pembatalan pemesanan.
5. Mengonfirmasi pembatalan indeks.	6. Membatalkan pemesanan.
A.5. Tidak mengonfirmasi pembatalan indeks.	
Alur Kejadian Alternatif	
Pencari indeks	Sistem
A.5 Pencari indeks tidak mengonfirmasi pembatalan pemesanan.	A.6 Sistem menampilkan daftar transaksi.



**Gambar 3.6 Diagram Aktivitas Melakukan Pembatalan Pemesanan Indekos**

#### **3.1.5.4 Melakukan Konfirmasi Pembayaran Indekos (UC-0004)**

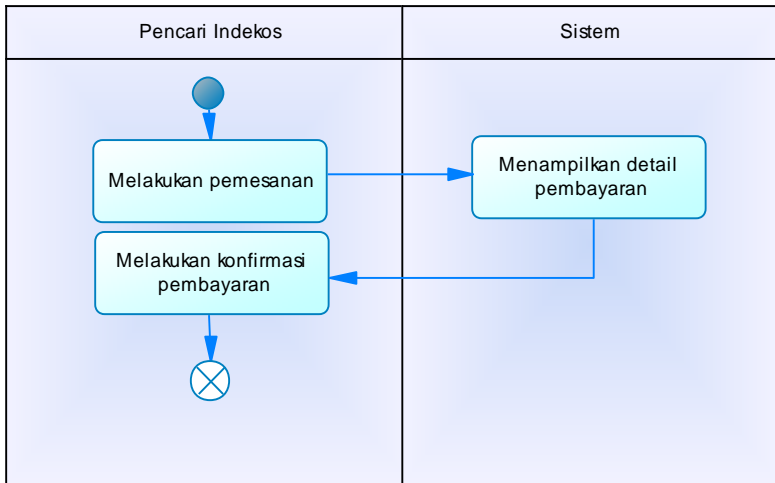
Pada kasus penggunaan ini, pencari indekos setelah melakukan pembayaran sesuai dengan harga yang tertera di aplikasi CariKos. Setelah melakukan pembayaran pencari indekos dapat melakukan konfirmasi pembayaran indekos yang sudah dibayarkan. Pencari indekos diwajibkan mengisi formulir konfirmasi pembayaran yang disediakan di aplikasi CariKos. Setelah melakukan pengisian formulir konfirmasi pembayaran, maka secara otomatis sistem akan membaca bahwa pencari indekos sudah melunasi pembayaran. Apabila pencari indekos tidak mengisi formulir konfirmasi pembayaran, maka sistem akan membaca bahwa pemesanan yang dilakukan pencari indeko belum dibayarkan atau belum lunas. Spesifikasi kasus penggunaan

melakukan pemesanan indeks dapat dilihat pada Tabel 3.6 dan diagram aktivitas pada Gambar 3.7.

**Tabel 3.6 Spesifikasi Kasus Penggunaan Melakukan Konfirmasi Pembayaran Indeks**

Kode	UC-0004
Nama	Melakukan Konfirmasi Pembayaran Indeks
Aktor	Pencari Indeks
Deskripsi	Pencari indeks melakukan konfirmasi pembayaran indeks yang sudah dipesan
Tipe	Fungsional
Kondisi Awal	Pencari indeks melakukan pemesanan
Kondisi Akhir	Pencari indeks melakukan konfirmasi pembayaran pemesanan indeks
Alur Kejadian Normal	
Pencari indeks	Sistem
1. Melakukan pemesanan.	2. Menampilkan detail pembayaran.
3. Melakukan konfirmasi pembayaran.	
Alur Kejadian Alternatif	
Pencari indeks	Sistem
Tidak ada.	Tidak ada.





**Gambar 3.7 Diagram Aktivitas Melakukan Pembayaran Indeks**

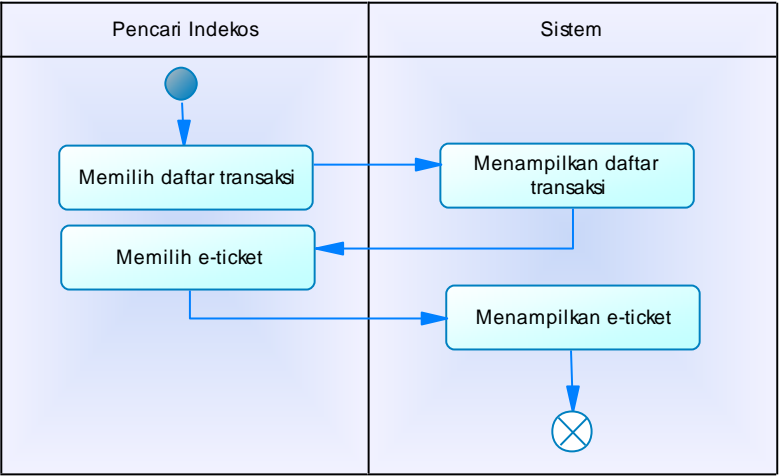
### 3.1.5.5 Mendapatkan *E-Ticket* (UC-0005)

Pada kasus penggunaan ini, pencari indeks mendapatkan *e-ticket* indeks yang sudah dibayar. Spesifikasi kasus penggunaan melakukan pemesanan indeks dapat dilihat pada Tabel 3.7 dan diagram aktivitas pada Gambar 3.8.

**Tabel 3.7 Spesifikasi Kasus Penggunaan Mendapatkan *E-Ticket***

Kode	UC-0005	
Nama	Melakukan Pembayaran Indeks	
Aktor	Pencari Indeks	
Deskripsi	Pencari indeks mendapatkan <i>e-ticket</i> indeks yang sudah dibayar	
Tipe	Fungsional	
Kondisi Awal	Pencari indeks memilih daftar transaksi	
Kondisi Akhir	Pencari indeks sudah mendapatkan <i>e-ticket</i>	
Alur Kejadian Normal		
Pencari indeks		Sistem
1. Memilih daftar transaksi.		

3. Memilih <i>e-ticket</i> .	2. Menampilkan daftar transaksi. 4. Menampilkan <i>e-ticket</i> .
Alur Kejadian Alternatif	
Pencari indekos	Sistem
Tidak ada.	Tidak ada.



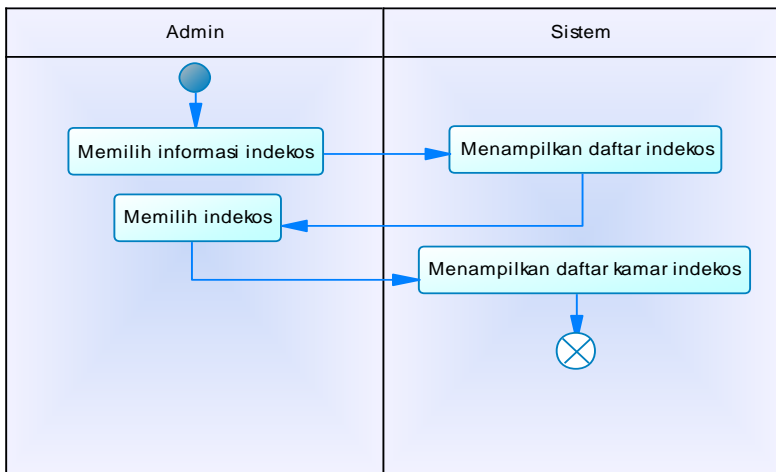
Gambar 3.8 Diagram Aktivitas Mendapatkan *E-Ticket*

3.1.5.6 Melihat Informasi Indekos (UC-0006)

Pada kasus penggunaan ini, admin dapat melihat informasi indekos yang terdaftar di CariKos. Informasi yang dapat dilihat adalah kamar dan indekos yang sudah terdaftar di CariKos. Spesifikasi kasus penggunaan melakukan pemesanan indekos dapat dilihat pada Tabel 3.8 dan diagram aktivitas pada Gambar 3.9.

**Tabel 3.8 Spesifikasi Kasus Penggunaan Melihat Informasi Indekos**

Kode	UC-0006
Nama	Melihat Kamar Indekos
Aktor	Admin
Deskripsi	Admin dapat melihat kamar indekos yang terdaftar di CariKos
Tipe	Fungsional
Kondisi Awal	Admin memilih informasi indekos
Kondisi Akhir	Sistem menampilkan daftar kamar indekos
Alur Kejadian Normal	
Admin	Sistem
1. Memilih informasi indekos.	2. Menampilkan daftar indekos.
3. Memilih indekos.	4. Menampilkan daftar kamar indekos.
Alur Kejadian Alternatif	
Admin	Sistem
Tidak ada.	Tidak ada.

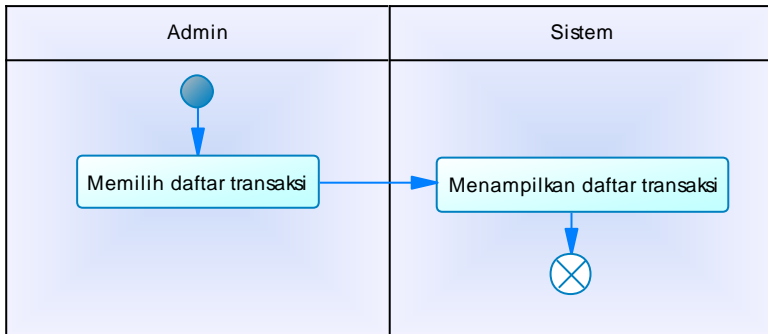
**Gambar 3.9 Diagram Aktivitas Melihat Informasi Indekos**

### 3.1.5.7 Melihat Transaksi (UC-0007)

Pada kasus penggunaan ini, admin dapat transaksi yang terjadi di CariKos. Spesifikasi kasus penggunaan melakukan pemesanan indeks dapat dilihat pada Tabel 3.9 dan diagram aktivitas pada Gambar 3.10.

**Tabel 3.9 Spesifikasi Kasus Penggunaan Melihat Transaksi**

Kode	UC-0007
Nama	Melihat Transaksi
Aktor	Admin
Deskripsi	Admin dapat melihat transaksi yang terjadi di CariKos
Tipe	Fungsional
Kondisi Awal	Admin memilih daftar transaksi
Kondisi Akhir	Sistem menampilkan daftar transaksi
Alur Kejadian Normal	
Admin	Sistem
1. Memilih daftar transaksi.	2. Menampilkan daftar transaksi.
Alur Kejadian Alternatif	
Admin	Sistem
Tidak ada.	Tidak ada.



**Gambar 3.10 Diagram Aktivitas Melihat Transaksi**

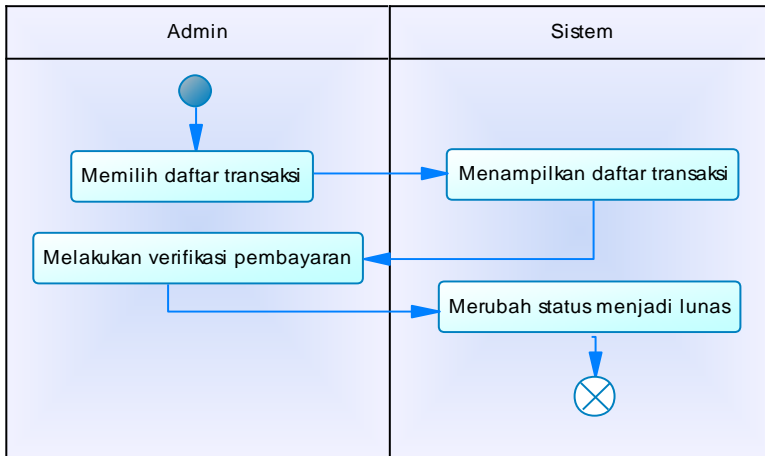
### 3.1.5.8 Verifikasi Pembayaran (UC-0008)

Pada kasus penggunaan ini, admin dapat verifikasi pembayaran yang dilakukan pencari indekos. Spesifikasi kasus penggunaan melakukan pemesanan indekos dapat dilihat pada Tabel 3.10 dan diagram aktivitas pada Gambar 3.11.

**Tabel 3.10 Spesifikasi Kasus Penggunaan Verifikasi Pemabayaran**

Kode	UC-0009
Nama	Verifikasi Pembayaran
Aktor	Admin
Deskripsi	Admin dapat verifikasi pembayaran yang dilakukan pencari indekos
Tipe	Fungsional
Kondisi Awal	Admin memilih daftar transaksi
Kondisi Akhir	Admin sudah verifikasi pemabayaran
Alur Kejadian Normal	
Admin	Sistem
1. Memilih daftar transaksi.	2. Menampilkan daftar transaksi.
3. Melakukan verifikasi pembayaran.	

	4. Merubah status menjadi lunas.
Alur Kejadian Alternatif	
Admin	Sistem
Tidak ada.	Tidak ada.



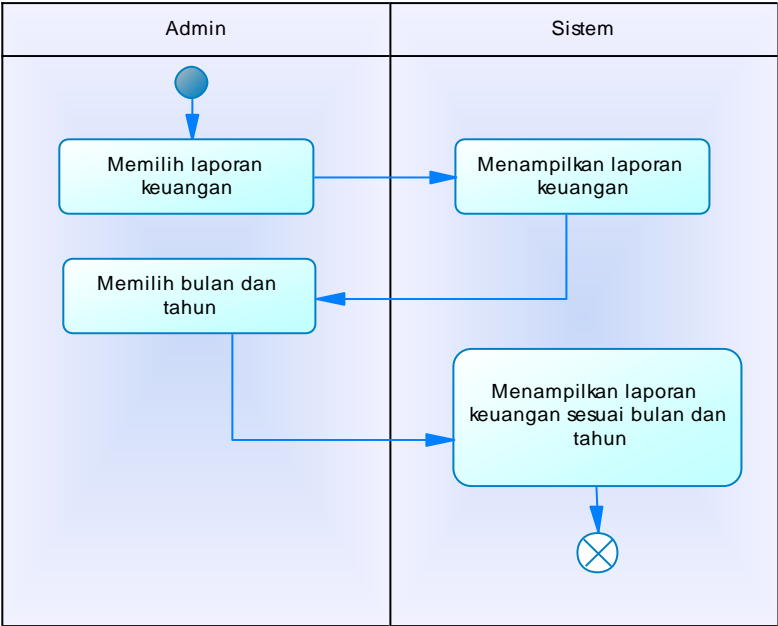
**Gambar 3.11 Diagram Aktivitas Verifikasi Pembayaran**

### 3.1.5.9 Melihat Laporan Keuangan (UC-0009)

Pada kasus penggunaan ini, admin dapat melihat laporan keuangan. Admin dapat memilih bulan dan tahun yang akan dilihat laporan keuangannya. Apabila pada bulan dan tahun yang dipilih tidak tersedia laporan keuangannya, maka sistem akan menampilkan bahwa data yang dicari tidak tersedia. Pada laporan keuangan ini juga admin dapat memantau keuangan dari aplikasi indekos. Dari laporan keuangan ini dapat dilihat keuntungan/kerugian yang dialami oleh aplikasi CariKos. Spesifikasi kasus penggunaan melakukan pemesanan indekos dapat dilihat pada Tabel 3.11 dan diagram aktivitas pada Gambar 3.12.

**Tabel 3.11 Spesifikasi Kasus Penggunaan Melihat Laporan Keuangan**

Kode	UC-0009
Nama	Melihat Laporan Keuangan
Aktor	Admin
Deskripsi	Admin dapat melihat laporan keuangan pada bulan tertentu dan tahun tertentu
Tipe	Fungsional
Kondisi Awal	Admin memilih laporan keuangan
Kondisi Akhir	Sistem menampilkan laporan keuangan
Alur Kejadian Normal	
Admin	Sistem
1. Memilih laporan keuangan.	2. Menampilkan laporan keuangan.
3. Memilih bulan dan tahun.	
	4. Menampilkan laporan keuangan sesuai dengan bulan dan tahun yang dipilih.
Alur Kejadian Alternatif	
Admin	Sistem
Tidak ada.	Tidak ada.



Gambar 3.12 Diagram Aktivitas Melihat Laporan Keuangan

3.1.5.10 Melihat Pemesanan (UC-0010)

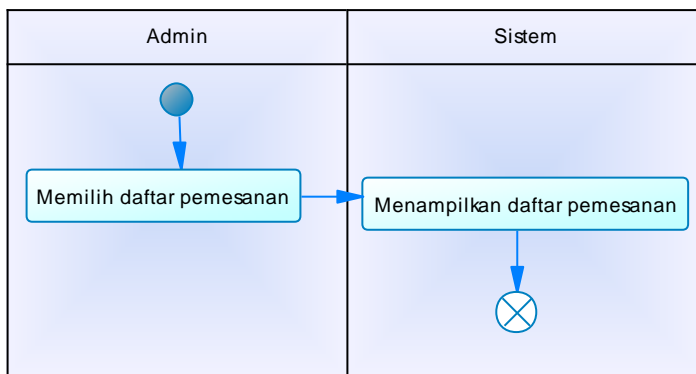
Pada kasus penggunaan ini, admin dapat melihat daftar pemesanan. Spesifikasi kasus penggunaan melakukan pemesanan indeks dapat dilihat pada Tabel 3.12 dan diagram aktivitas pada Gambar 3.13.

Tabel 3.12 Spesifikasi Kasus Penggunaan Melihat Pemesanan

Kode	UC-0010
Nama	Melihat Pemesanan
Aktor	Admin
Deskripsi	Admin dapat melihat daftar pemesanan
Tipe	Fungsional
Kondisi Awal	Admin memilih daftar pemesanan
Kondisi Akhir	Sistem menampilkan daftar pemesanan



Alur Kejadian Normal	
Admin	Sistem
1. Memilih daftar pemesanan.	2. Menampilkan daftar pemesanan.
Alur Kejadian Alternatif	
Admin	Sistem
Tidak ada.	Tidak ada.



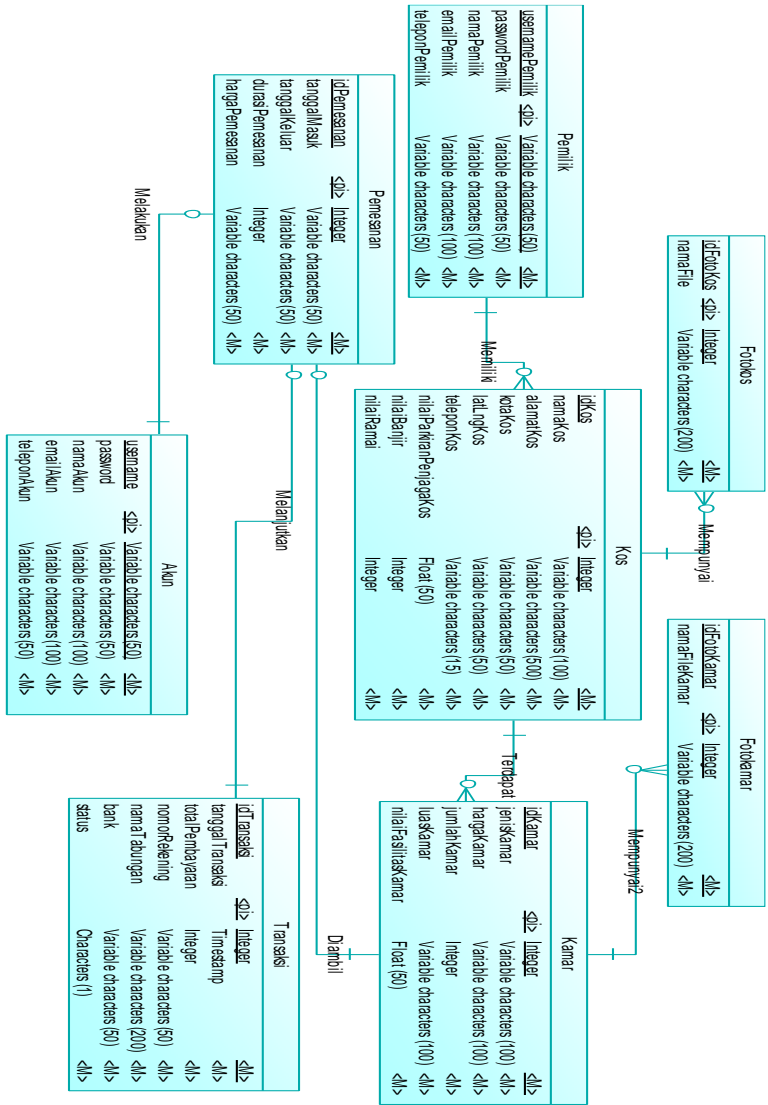
**Gambar 3.13 Diagram Aktivitas Melihat Pemesanan**

## 3.2 Perancangan Sistem

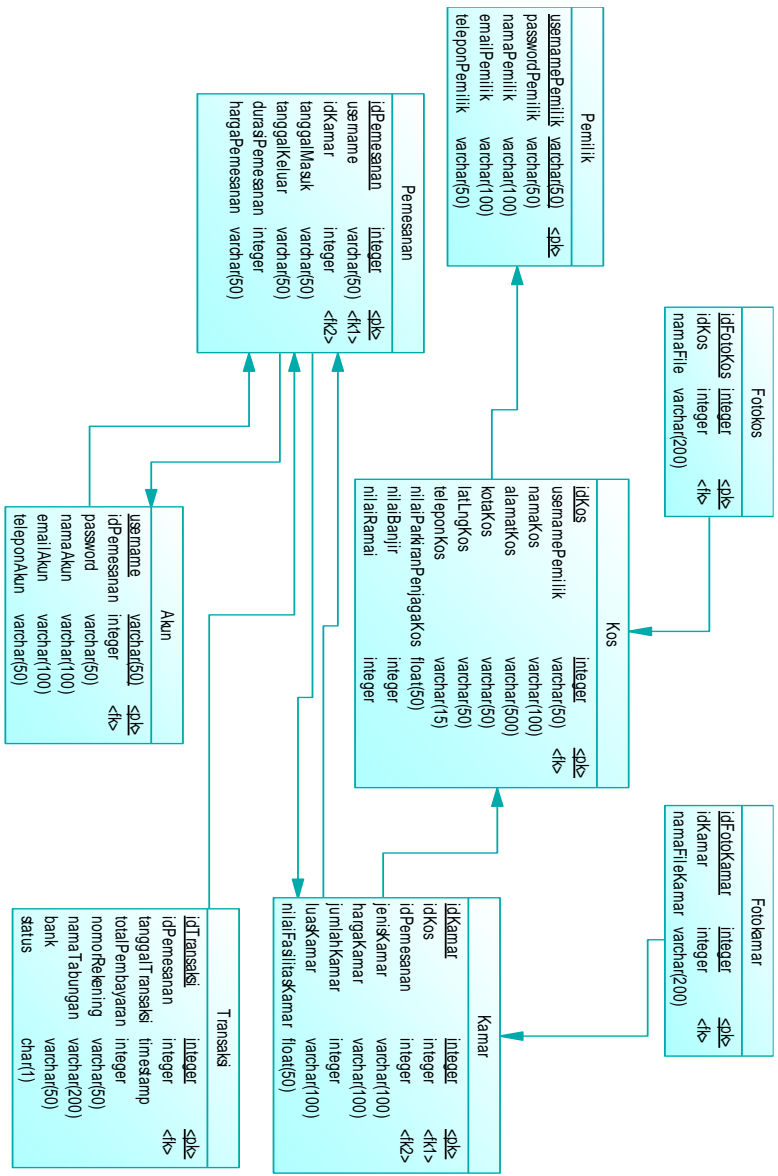
Tahap ini meliputi perancangan basis data, tampilan antarmuka, dan perancangan arsitektur sistem yang diharapkan dapat memenuhi tujuan dari pengembangan aplikasi ini.

### 3.2.1 Perancangan Basis Data

Pada subbab ini dijelaskan mengenai perancangan basis data yang dalam hal ini digunakan untuk menyimpan data indeks, kamar dan transaksi untuk menentukan harga indeks. Gambaran perancangan basis data dapat dilihat pada Gambar 3.14 dan Gambar 3.15.



Gambar 3.14 Conceptual Data Model



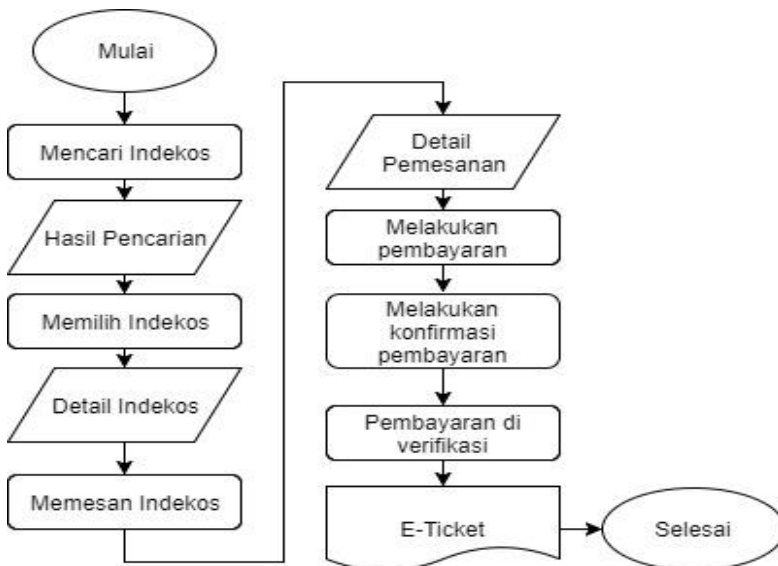
Gambar 3.15 Physical Data Model

### 3.2.2 Perancangan Algoritma

Pada subbab ini dijelaskan mengenai perancangan algoritma yang akan digunakan dalam aplikasi CariKos. Terdapat dua perancangan pada modul pemesanan indeksos dan penentuan harga indeksos.

#### 3.2.2.1 Perancangan Algoritma Modul Pemesanan

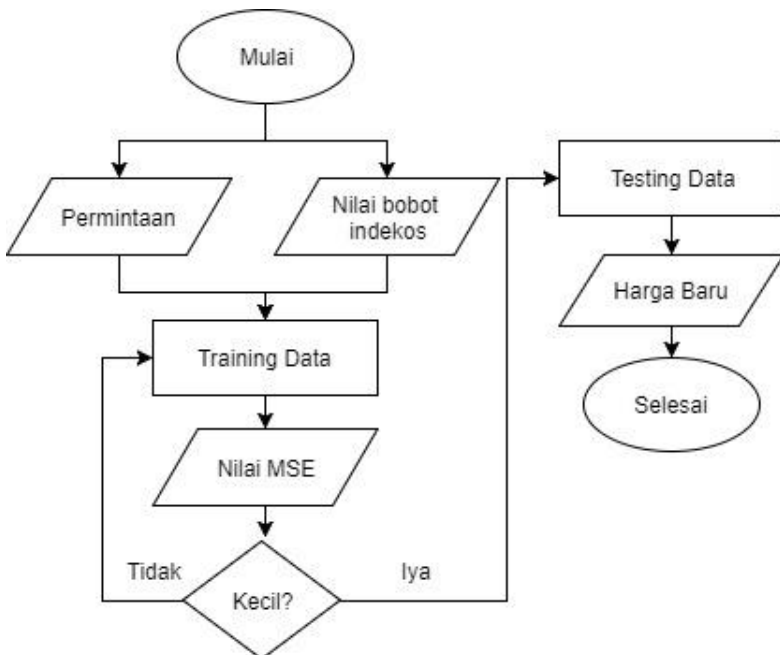
Modul pemesanan ini dibagi dalam beberapa proses, yaitu dimulai dari pencarian yang dilakukan oleh pencari pada aplikasi indeksos, lalu melakukan pemesanan indeksos yang sudah dipilih oleh pencari indeksos, melakukan pembayaran dari pemesanan yang dilakukan pencari, melakukan konfirmasi pembayaran yang sudah dilakukan, dan terakhir menunggu verifikasi pembayaran dari admin CariKos. Pada Gambar 3.16 dapat dilihat diagram alir dari modul pemesanan.



Gambar 3.16 Diagram Alir Pemesanan

### 3.2.2.2 Perancangan Algoritma Modul Penentuan Harga

Pada modul penentuan harga ini akan dijelaskan proses penentuan harga dari metode back propagation. Terdapat dua proses utama pada metode back propagation yaitu, *training* data dan *testing* data. Diagram alir back propagation dapat dilihat pada Gambar 3.17.



Gambar 3.17 Diagram Alir Metode Back Propagation

#### A. Perancangan Algoritma pada *Training* Data

Pada perancangan algoritma training data akan dijelaskan proses data di-*training* hingga mendapatkan nilai MSE yang kecil. Proses *training* data diawali dengan melakukan normalisasi dua *input* terlebih dahulu. Metode normalisasi yang digunakan adalah

metode *min-max*. Berikut ini merupakan rumus dari metode *min-max*:

$$x' = \frac{0.8(x - a)}{b - a} + 0.1 \quad (1)$$

Keterangan :

$x'$  = hasil normalisasi  
 $x$  = data awal  
 $a$  = data minimal  
 $b$  = data maksimal

Dari rumus (1) dapat dilihat bahwa data awal merupakan *input* sebelum dinormalisasi, data minimal merupakan nilai *input* yang paling kecil, dan data maksimal merupakan nilai *input* yang paling besar.

Setelah melakukan normalisasi data, menginisiasi *weight* awal antar *node* dengan nilai *random*. Nilai *random* tersebut memiliki rentang nilai dari -1 hingga 1. Lalu melakukan perhitungan nilai pada hidden layer dengan menggunakan rumus berikut:

$$z_{net\ j} = v_{j0} + \sum_{i=1}^n x_i \cdot v_{ji} \quad (2)$$

Keterangan:

$z_{net\ j}$  = nilai untuk menghitung *hidden layer*  
 $v_{j0}$  = bobot bias antara *input layer* dengan *hidden layer*  
 $x_i$  = nilai *input layer* ke- $i$   
 $v_{ji}$  = bobot antara *input layer* dengan *hidden layer*

Hasil dari perhitungan pada rumus (2) dilakukan aktivasi agar nilai hasilnya tetap berada di rentang nilai tertentu. Untuk melakukan aktivasi digunakan fungsi sigmoid agar rentang nilai berada di rentang 0 hingga 1. Rumus sigmoid yang digunakan:

$$z_j = f(z_{net_j}) = \frac{1}{1 + \exp(-z_{net_j})} \quad (3)$$

Keterangan:

$z_j$  = nilai *hidden layer*

$z_{net_j}$  = nilai untuk menghitung *hidden layer*

Selanjutnya menghitung nilai *output*, yang mempunyai rumus sebagai berikut:

$$y_{net_k} = w_{k0} + \sum_{j=1}^p z_j \cdot w_{kj} \quad (4)$$

Information:

$y_{net_k}$  = net masukan unit k

$w_{k0}$  = nilai penimbang sambungan pada bias untuk unit  $y_k$

$z_j$  = nilai aktivasi dari unit  $z_j$

$w_{kj}$  = nilai penimbang sambungan dari  $z_{ij}$  ke unit  $y_k$

Hasil dari perhitungan rumus (4) dilakukan aktivasi agar nilainya mempunyai rentang tertentu. Maka digunakan aktivasi fungsi sigmoid agar nilai berada pada rentang 0 hingga 1. Berikut ini rumus aktivasi fungsi sigmoid:

$$y_k = f(y_{net_k}) = \frac{1}{1 + \exp(-y_{net_k})} \quad (5)$$

Keterangan:

$y_k$  = nilai *output*

$y_{net_k}$  = net masukan unit k

Setelah melakukan perhitungan dengan rumus (4) maka dilakukan perhitungan kesalahan pada setiap *output* dengan rumus sebagai berikut:

$$\delta_k = (t_k - y_k)f'(y_{net_k}) = (t_k - y_k)y_k(1 - y_k) \quad (6)$$

Keterangan:

$\delta_k$  = nilai kesalahan pada *output layer*

$t_k$  = nilai target data

$y_k$  = nilai *output*

$y_{net_k}$  = net masukan unit k

$\delta_k$  merupakan nilai kesalahan yang dipakai untuk merubah bobot *layer* dibawahnya. Rumus yang digunakan adalah sebagai berikut:

$$\Delta w_{kj} = \alpha \cdot \delta_k \cdot z_j \quad (7)$$

Keterangan:

$\Delta w_{kj}$  = selisih antara  $w_{kj}(t)$  dengan  $w_{kj}(t + 1)$

$\alpha$  = nilai *learning rate*

$\delta_k$  = nilai kesalahan *output layer*

$z_j$  = nilai aktivasi dari unit  $z_j$

Selanjutnya menghitung nilai kesalahan pada *hidden layer* berdasarkan kesalahan pada setiap *hidden layer*  $z_j$  dengan rumus sebagai berikut:

$$\delta_{net_j} = \sum_{k=1}^m \delta_k \cdot w_{kj} \quad (8)$$

Keterangan:

$\delta_{net_j}$  = nilai untuk menghitung nilai kesalahan pada *hidden layer*

$\delta_k$  = nilai kesalahan pada *output layer*

$w_{kj}$  = bobot dari  $z_{ij}$  ke  $y_k$

Nilai kesalahan pada *hidden layer* dapat dihitung menggunakan rumus:

$$\delta_j = \delta_{net_j} \cdot f'(z_{net_j}) = \delta_{net_j} \cdot z_j (1 - z_j) \quad (9)$$

Information:

$\delta_{net_j}$  = nilai untuk menghitung nilai kesalahan pada *hidden layer*

$z_{net_j}$  = net masukan dari unit j



$z_j$  = nilai aktivasi dari unit  $z_j$

Setelah menghitung nilai kesalahan pada *hidden layer* maka dihitung perubahan bobot  $v_{ji}$  dengan menggunakan rumus:

$$\Delta v_{ji} = \alpha \cdot \delta_j \cdot x_i \quad (10)$$

Information:

$\Delta v_{ji}$  = selisih antara  $v_{ji}$  (t) dengan  $v_{ji}$  (t + 1)

$\alpha$  = nilai *learning rate*

$\delta_j$  = nilai kesalahan pada *hidden layer*

$x_i$  = unit ke-i pada *input layer*

Terakhir melakukan *update* pada setiap *weight*. Untuk menghitung perubahan nilai *weight* antara *hidden layer* dengan *output layer* menggunakan rumus:

$$w_{kj}(\text{new}) = w_{kj}(\text{old}) + \Delta w_{kj} \quad (11)$$

Keterangan:

$w_{kj}$  = bobot dari  $z_{ij}$  ke  $y_k$  unit

$\Delta w_{kj}$  = selisih antara  $w_{kj}$  (t) dengan  $w_{kj}$  (t + 1)

Untuk menghitung perubahan nilai *weight* antara *input layer* dengan *hidden layer* menggunakan rumus:

$$v_{ji}(\text{new}) = v_{ji}(\text{old}) + \Delta v_{ji} \quad (12)$$

Keterangan:

$v_{ji}$  = bobot dari  $x_i$  unit ke  $z_i$  unit

$\Delta v_{ji}$  = selisih antara  $v_{ji}$  (t) dengan  $w_{ij}$  (t + 1)

Setelah melakukan *update* pada setiap nilai *weight* maka dilakukan perhitungan nilai MSE. *Mean Square Error* (MSE) merupakan metode yang digunakan untuk menguji kesalahan pada sistem. Rumus perhitungan MSE adalah sebagai berikut:

$$MSE = \frac{1}{k} \sum_{k=1}^n (y_k - t_k)^2 \quad (13)$$

Keterangan:

$n$  = jumlah data  
 $y_k$  = nilai *output* data  
 $t_k$  = nilai target data

*Training* data dilakukan terus menerus hingga mendapatkan nilai MSE dibawah 0.0728. Jika nilai MSE sudah dibawah 0.0728 maka dilakukan *testing* data untuk mendapatkan harga baru.

## B. Perancangan Algoritma pada *Testing* Data

Setelah *training* data sudah selesai, maka dilakukan *testing* data agar mendapat harga yang baru. *Testing* data dilakukan dengan proses perhitungan rumus (1) hingga rumus (5) pada *training* data. Nilai *output* yang didapatkan dilakukan denormalisasi terlebih dahulu dengan menggunakan metode *min-max* agar menghasilkan harga baru. Rumus denormalisasi metode *min-max* adalah sebagai berikut:

$$x = \frac{(x' - 0.1)(x_{max} - x_{min})}{0.8} + x_{min}$$

Keterangan :

$x$  = data asli  
 $x'$  = data normalisasi  
 $x_{max}$  = data maksimum asli  
 $x_{min}$  = data minimum asli

Nilai maksimum diambil dari harga awal ditambah lima persen dari harga asli, sedangkan nilai minimum diambil dari harga awal dikurangi dua setengah persen dari harga awal.

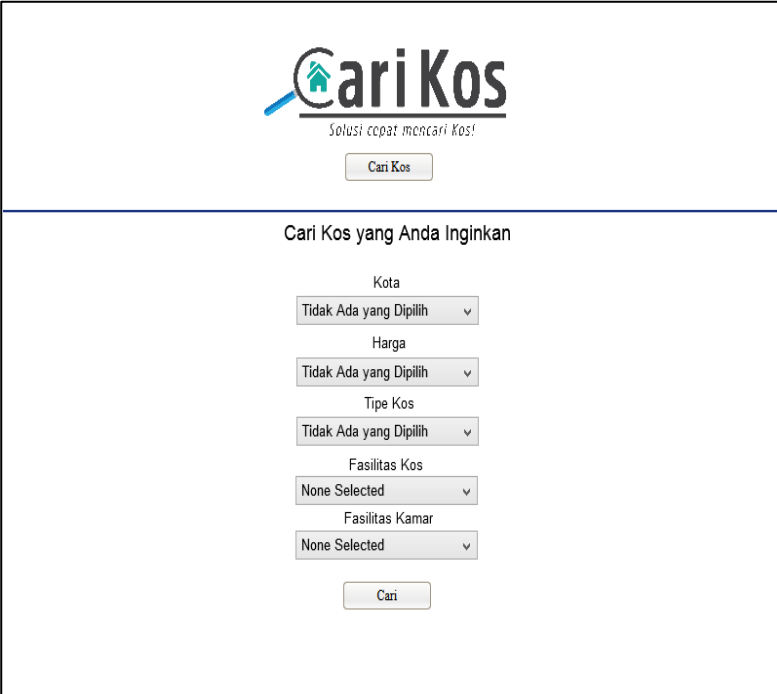
Hasil dari denormalisasi akan menghasilkan harga yang digunakan pada aplikasi CariKos. Penentuan harga ini diharapkan membuat harga yang *fluktuatif* seperti harga sewa hotel yang berbeda setiap bulannya sesuai dengan permintaan dari pasar.

### 3.2.3 Perancangan Tampilan Antarmuka

Subbab ini menjelaskan bagaimana rancangan antarmuka yang akan berinteraksi secara langsung dengan pengguna.

#### 3.2.3.1 Perancangan Halaman Beranda

Halaman ini merupakan halaman utama yang menampilkan menu cari kos. Dari menu ini pencari indekos dapat mencari indekos sesuai dengan keinginannya dengan mengisi *form* yang tersedia. Rancangan halaman beranda dapat dilihat pada Gambar 3.17.



**Cari Kos**  
Solusi cepat mencari Kos!

Cari Kos

Cari Kos yang Anda Inginkan

Kota  
Tidak Ada yang Dipilih ▼

Harga  
Tidak Ada yang Dipilih ▼

Tipe Kos  
Tidak Ada yang Dipilih ▼

Fasilitas Kos  
None Selected ▼

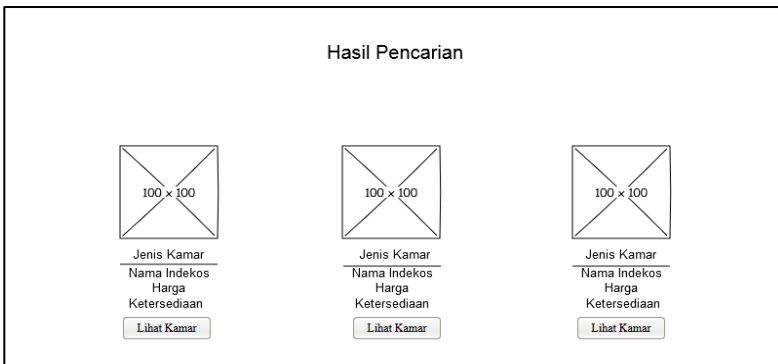
Fasilitas Kamar  
None Selected ▼

Cari

**Gambar 3.18 Rancangan Halaman Beranda**

### 3.2.3.2 Perancangan Halaman Hasil Pencarian

Pada halaman ini ditampilkan hasil pencarian yang dilakukan pengguna pada halaman beranda. Hasil pencarian yang diberikan akan menampilkan beberapa informasi, seperti foto, jenis kamar, nama kos, harga kamar, dan ketersediaan kamar. Rancangan halaman dapat dilihat pada Gambar 3.18



**Gambar 3.19 Rancangan Halaman Hasil Pencarian**

### 3.2.3.3 Perancangan Halaman Detail Kamar dan Indekos

Pada halaman ini ditampilkan detail kamar dan indekos. Detail kamar dan indekos ini memberikan informasi dari detail kamar dan indekos yang sudah kita pilih. Rancangan halaman dapat dilihat pada Gambar 3.19.

### Jenis Kamar - Nama Indeks

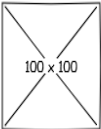
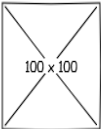
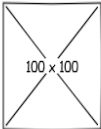
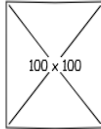

#### Detail Kamar

Harga kamar	<Harga kamar>
Jumlah Kamar yang tersedia	<Jumlah kamar>
Luas kamar	<Luas kamar>
Fasilitas kamar	<Fasilitas kamar>

#### Detail Kos

Alamat Kos	<Alamat Kos>
Telepon Kos	<Telepon Kos>
Tipe Kos	<Tipe Kos>
Penjaga Kos	<Ya/Tidak>
Luas Parkiran	<Luas Parkiran>
Fasilitas Kos	<Fasilitas Kos>

#### Foto Kamar dan Kos

**Gambar 3.20 Rancangan Detail Kamar dan Indeks**

### 3.2.3.4 Perancangan Halaman *Form* Pemesanan Kamar

Pada halaman ini ditampilkan *form* pemesanan kamar. *Form* pemesanan kamar ini berisi data pemesan yang berupa nama pemesan, nomor telepon pemesan, email pemesan, tanggal masuk, dan tanggal keluar. Selain data pemesan terdapat juga rincian kamar dan kode promo. Rancangan halaman dapat dilihat pada Gambar 3.20.

Pemesanan Kamar		
Data Pemesan		Rincian Kamar
Nama Pemesan	<input type="text"/>	Nama Indekos <Nama Indekos>
Nomor Telepon	<input type="text"/>	Jenis Kamar <Jenis Kamar>
Email Pemesan	<input type="text"/>	Harga Kamar <Harga Kamar>
Tanggal Masuk	<input type="text"/>	Tipe Kos <Tipe Kos>
Tanggal Keluar	<input type="text"/>	
Kode Promo		
Kode	<input type="text"/>	
<input type="button" value="Lanjutkan Pemesanan"/>		

**Gambar 3.21 Rancangan Halaman Form Pemesanan Kamar**

### **3.2.3.5 Perancangan Halaman Detail Pemesanan**

Pada halaman ini ditampilkan detail pemesanan dan rincian pembayaran. Detail pemesanan berisi nama pemesan, nomor telepon pemesan, email pemesan, nama indekos, tipe kos, jenis kamar, harga kamar, tanggal masuk, dan tanggal keluar. Rincian biaya berisi harga kamar, lama sewa, total yang harus dibayar. Rancangan halaman dapat dilihat pada Gambar 3.21.

Pemesanan Kamar

Detail Pemesanan

Nama Pemesan  
<Nama Pemesan>

Nomor Telepon Pemesan  
<Nomor Telepon Pemesan>

Nama Kos  
<Nama Kos>

Jenis Kamar  
<Jenis Kamar>

Tanggal Masuk  
<Tanggal Masuk>

Email Pemesan  
<Email Pemesan>

Tipe Kos  
<Tipe Kos>

Harga Kamar  
<Harga Kamar>

Tanggal Keluar  
<Tanggal Keluar>

Rincian Pembayaran

<Nama Indekos>

<Jenis Kamar>   <Harga Kamar>

Durasi Sewa   <Durasi Sewa>

Total Pembayaran   <Total Pembayaran>

Lanjut ke Pembayaran

**Gambar 3.22 Rancangan Halaman Detail Pemesanan**

### **3.2.3.6 Perancangan Halaman Pembayaran**

Pada halaman ini ditampilkan rincian pembayaran dan pemberitahuan rekening tujuan untuk melakukan transaksi. Rancangan halaman dapat dilihat pada Gambar 3.22.

**Pembayaran**

**Rincian Pembayaran**

ID Transaksi	<ID Transaksi>
Total Pembayaran	<Total Pembayaran>

**Transfer ke**

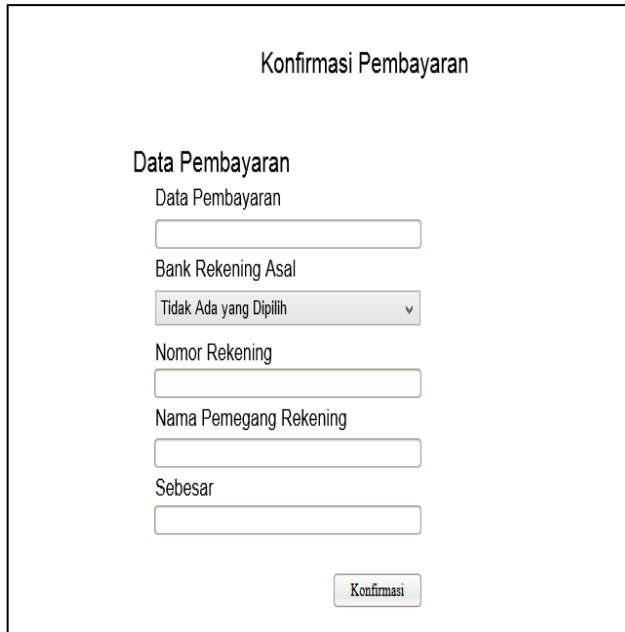
<Nama Bank>	<Nomor Rekening>
Atas Nama	CariKos

**Gambar 3.23 Rancangan Halaman Pembayaran**

### **3.2.3.7 Perancangan Halaman Konfirmasi Pembayaran**

Pada halaman ini ditampilkan *form* data pembayaran yang berisi id transaksi, bank rekening asal, nomor rekening, nama pemegang rekening, dan besar transfer.





Konfirmasi Pembayaran

Data Pembayaran

Data Pembayaran

Bank Rekening Asal

Tidak Ada yang Dipilih

Nomor Rekening

Nama Pemegang Rekening

Sebesar

Konfirmasi

**Gambar 3.24 Rancangan Halaman Konfirmasi Pembayaran**

### **3.2.3.8 Perancangan Halaman Daftar Transaksi di sisi Pencari Indeks**


Pada halaman ini ditampilkan daftar transaksi di aplikasi CariKos. Daftar transaksi berisi id transaksi, nama kos, jenis kamar, durasi sewa, total harga, status, dan pembatalan. Status berisi belum bayar jika belum melakukan pembayaran, berisi batal jika membatalkan pemesanan, dan berisi tombol *e-ticket* yang berfungsi untuk mendapatkan *e-ticket* jika sudah melakukan pembayaran. Pembatalan berisi tombol batal jika pembayaran sudah lunas atau sudah dilakukan pembatalan maka tombol batal akan dimatikan dan sebaliknya jika belum melakukan pembayaran dan pembatalan maka tombol batal akan bisa di tekan.

Daftar Transaksi						
ID Transaksi	Nama Kos	Jenis Kamar	Durasi Sewa	Total Harga	Status	Pembatalan
1	<Nama Kos>	<Jenis Kamar>	<Durasi Sewa>	<Total Harga>	Belum Bayar	<button>Batal</button>
2	<Nama Kos>	<Jenis Kamar>	<Durasi Sewa>	<Total Harga>	Batal	<button>Batal</button>
3	<Nama Kos>	<Jenis Kamar>	<Durasi Sewa>	<Total Harga>	E-Ticket	<button>Batal</button>

**Gambar 3.25 Rancangan Halaman Daftar Transaksi di sisi Pencari Indeks**

### 3.2.3.9 Perancangan Halaman Utama Admin

Pada halaman ini ditampilkan halaman utama dari sisi admin. Halaman ini berisi ringkasan dari jumlah kamar, jumlah indeks, jumlah pemilik, dan laporan keuntungan.



- Dashboard
- Master Data
- Indekos
- Transaksi
- Portal
- Promo
- Daftar Promo
- Tambah Promo
- Laporan Keuangan

### Ringkasan

Indekos 3	Kamar 3	Pemilik 3	Keuntungan 1.000.000
--------------	------------	--------------	-------------------------

**Gambar 3.26 Rancangan Halaman Home Admin**

### 3.2.3.10 Perancangan Halaman Indekos

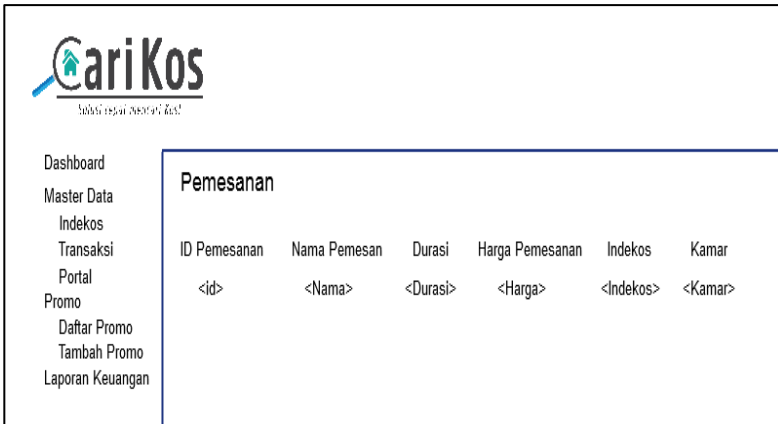
Pada halaman ini ditampilkan halaman indeks dari sisi admin. Halaman ini berisi nama indeks, alamat indeks, nomor telepon, jumlah kamar, dan nama pemilik.



**Gambar 3.27 Rancangan Halaman Indeks**

### 3.2.3.11 Perancangan Halaman Pemesanan


Pada halaman ini ditampilkan halaman daftar pemesanan dari sisi admin. Halaman ini berisi id pemesanan, nama pemesan, durasi pemesanan, harga pemesanan, nama indeks, dan jenis kamar.



**Gambar 3.28 Rancangan Halaman Pemesanan**

### 3.2.3.12 Perancangan Halaman Transaksi

Pada halaman ini ditampilkan halaman daftar transaksi dari sisi admin. Halaman ini berisi id transaksi, id pemesanan, tanggal pembayaran, bank asal, nomor rekening, nama pemilik, dan status pembayaran. Jika pencari indekos sudah melakukan konfirmasi transaksi maka akan muncul tombol verifikasi untuk melakukan verifikasi pembayaran.



**CariKos**  
Solusi tempat tinggal yang terbaik

Dashboard  
Master Data  
Indekos  
Transaksi  
Portal  
Promo  
Daftar Promo  
Tambah Promo  
Laporan Keuangan

### Transaksi

ID Transaksi	ID Pemesanan	Tanggal Pembayaran	Total Pembayaran	Bank Asal	Nomor Rekening	Nama Pemilik	Status
<ID Transaksi>	<ID Pemesanan>	<Tanggal>	<Harga>	<bank>	<nomor rekening>	<nama>	lunas
<ID Transaksi>	<ID Pemesanan>	<Tanggal>	<Harga>	<bank>	<nomor rekening>	<nama>	belum bayar
<ID Transaksi>	<ID Pemesanan>	<Tanggal>	<Harga>	<bank>	<nomor rekening>	<nama>	<a href="#">Verifikasi</a>

**Gambar 3.29 Rancangan Halaman Transaksi**

### 3.2.3.13 Perancangan Halaman Laporan Keuangan

Pada halaman ini ditampilkan halaman laporan keuangan dari sisi admin. Halaman ini berisi laporan keuangan perbulan dari CariKos.



**CariKos**  
Solusi tempat tinggal yang terbaik

Dashboard  
Master Data  
Indekos  
Transaksi  
Portal  
Promo  
Daftar Promo  
Tambah Promo  
Laporan Keuangan

### Laporan Keuangan

Bulan :  Tahun :

ID Transaksi	Tanggal	Harga Asli	Harga Baru	Total
<ID Transaksi>	<Tanggal>	<Harga Asli>	<Harga Baru>	<total>
<ID Transaksi>	<Tanggal>	<Harga Asli>	<Harga Baru>	<total>
<ID Transaksi>	<Tanggal>	<Harga Asli>	<Harga Baru>	<total>

**Gambar 3.30 Rancangan Halaman Laporan Keuangan**

*[Halaman ini sengaja dikosongkan]*

## **BAB IV IMPLEMENTASI**

Bab ini membahas implementasi dari analisis dan perancangan sistem yang telah dibahas pada Bab III. Namun dalam penerapannya, rancangan tersebut dapat mengalami perubahan sedikit sewaktu-waktu apabila dibutuhkan.

### **4.1 Lingkungan Implementasi**

Dalam implementasinya, lingkungan yang digunakan sama seperti yang dituliskan pada rancangan, yakni menggunakan beberapa perangkat pendukung sebagai berikut.

#### **4.1.1 Lingkungan Implementasi Perangkat Keras**

Perangkat keras yang digunakan dalam pengembangan aplikasi adalah sebagai berikut:

Jenis : Komputer  
Tipe : Lenovo 10093  
Prosesor : Intel® Core™ i3-3240 CPU (3.40 GHz)  
Memori/RAM : 8 GB

#### **4.1.2 Lingkungan Implementasi Perangkat Lunak**

Perangkat lunak yang digunakan dalam implementasi aplikasi ini adalah sebagai berikut:

- Microsoft Windows 8.1 Pro sebagai sistem operasi.
- MySQL untuk mengimplementasikan rancangan basis data.
- SublimeText 3 sebagai *text editor*.
- Apache sebagai *web server*.
- Pencil untuk merancang tampilan antarmuka.
- Sybase Power Designer versi 15 untuk merancang basis data, *Activity Diagram*, *Use Case Diagram*.

## 4.2 Implementasi Tampilan Antarmuka

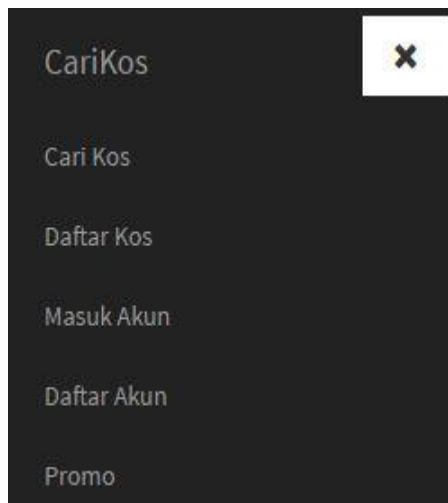
Pada subbab ini akan dijelaskan tentang implementasi antarmuka sistem yang menjadi bagian terluar sekaligus bagian yang akan berinteraksi langsung dengan pengguna.

### 4.2.1 Implementasi Halaman Beranda

Pada halaman ini terdapat dua buah menu, yaitu menu cari kos dan daftar kos seperti yang ditampilkan pada Gambar 4.2. Pada menu cari kos diberikan formulir pencarian yang berisikan kota, harga, tipe kos, fasilitas kos, fasilitas kamar, dan pilihan jurusan.

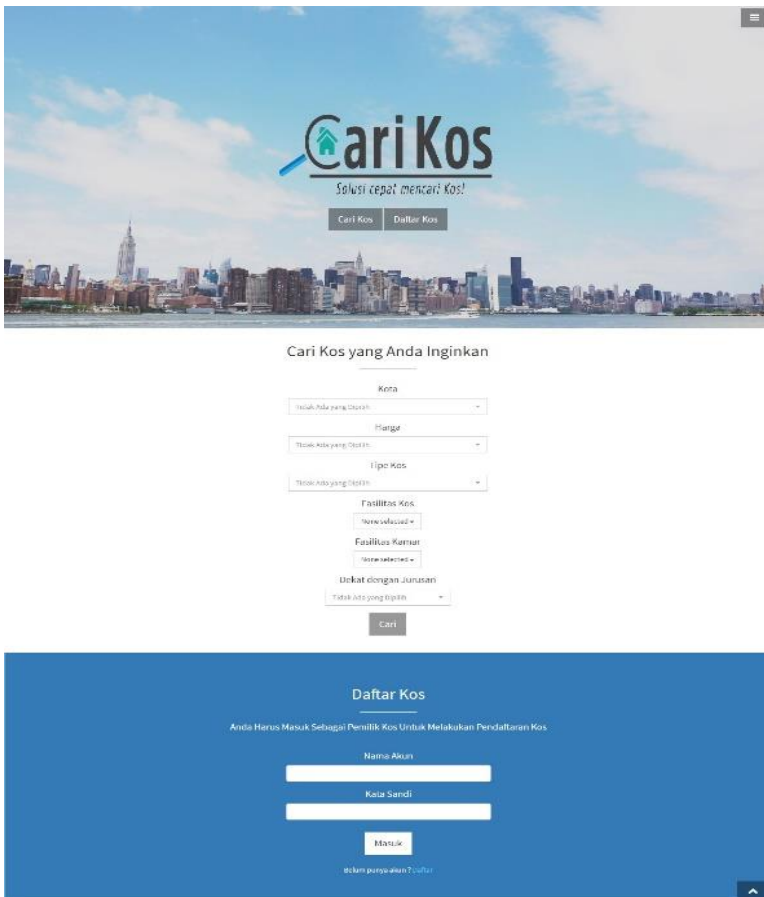
Pada menu daftar kos disajikan formulir untuk masuk ke akun pemilik. Karena untuk melakukan pendaftaran kos diharuskan masuk ke akun pemilik terlebih dahulu.

Di halaman ini pula diberikan *navigation bar* yang disajikan dengan sebuah tombol pada halaman pojok kanan atas. Menu-menu pada *navigation bar* (Gambar 4.1) berupa cari kos, daftar kos, daftar akun, masuk akun, dan promo.



**Gambar 4.1 Tampilan Navigation Bar Halaman Beranda**





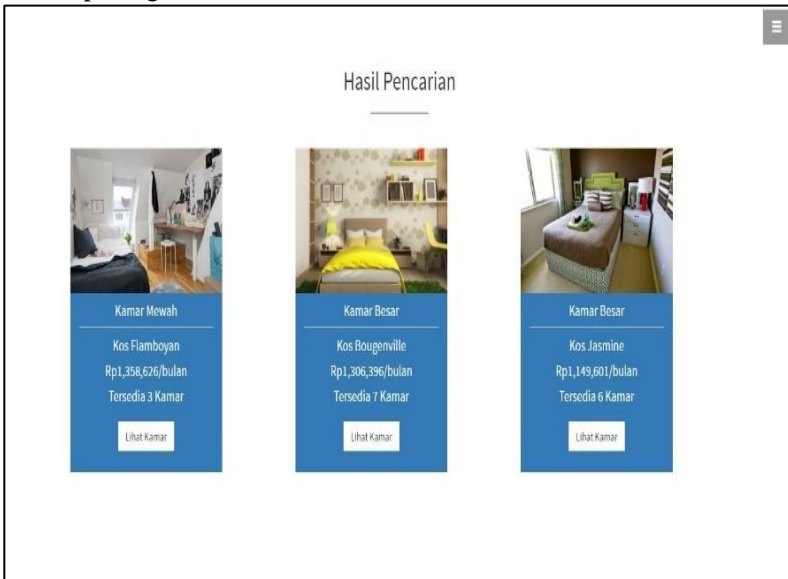
The image shows two screenshots of the 'Cari Kos' website. The top screenshot is the homepage with a city skyline background. It features the 'Cari Kos' logo with a magnifying glass icon and the tagline 'Solusi cepat mencari Kos!'. Below the logo are two buttons: 'Cari Kos' and 'Daftar Kos'. A section titled 'Cari Kos yang Anda Inginkan' contains several dropdown menus for filtering search results: 'Kota' (with a hint 'Tidak Ada yang dipilih'), 'Range' (with a hint 'Tidak Ada yang dipilih'), 'Tipe Kos' (with a hint 'Tidak Ada yang dipilih'), 'Fasilitas Kos' (with options 'Tidak ada fasilitas', 'Fasilitas Kamar', and 'Tidak ada fasilitas'), and 'Dekat dengan Juranan' (with a hint 'Tidak Ada yang dipilih'). A 'Cari' button is at the bottom of this section. The bottom screenshot is the 'Daftar Kos' (Register as Landlord) page, which has a solid blue background. It includes the title 'Daftar Kos', a sub-header 'Anda Harus Masuk Sebagai Pemilik Kos Untuk Melakukan Pendaftaran Kos', and two input fields for 'Nama Akun' and 'Kata Sandi'. A 'Mau' button is at the bottom, along with a small link 'atau punya akun? klik di sini'.

**Gambar 4.2 Tampilan Halaman Beranda**

#### **4.2.2 Implementasi Halaman Hasil Pencarian**

Halaman ini menampilkan hasil pencarian cari kos yang dilakukan pada halaman beranda. Hasil pencarian ditampilkan dalam sebuah *item* yang berisikan foto kamar, jenis kamar, nama


kos, dan jumlah kamar yang tersedia. Tampilan antarmuka dapat dilihat pada gambar 4.3.



**Gambar 4.3 Tampilan Halaman Hasil Pencarian**

#### **4.2.3 Implementasi Halaman Detail Kamar dan Indekos**

Halaman ini menampilkan detail kamar dan indekos yang sudah dipilih. Detail kamar dan indekos yang ditampilkan berisi harga kamar, jumlah kamar yang tersedia, luas kamar, fasilitas kamar, foto kamar, alamat indekos, nomor telepon indekos, tipe indekos, ada atau tidaknya penjaga indekos, luas parkir indekos, fasilitas indekos, foto indekos, dan informasi tempat-tempat penting sekitar indekos. Tampilan antar muka dapat dilihat pada gambar 4.4.



[Cari Kos](#)
[Transaksi ▾](#)
[Promo](#)
[Binowjani ▾](#)

### Kamar Besar - Kos Bougenville

Pesan Kamar



#### Detail Kamar

Harga Kamar	Rp1,308,841/bulan
Jumlah Kamar yang Tersedia	5 Kamar
Luas Kamar (Dalam m)	6 X 4
Fasilitas Kamar	Kasur Lemari Meja dan Kursi

#### Detail Kos

Alamat Kos	Jalan Teknik Komputer No.10, Keputih, Kota Surabaya, Jawa Timur, Indonesia
Telepon Kos	0314514204
Tipe Kos	Pria
Penjaga Kos	Ya
Luas Parkiran (dalam m <sup>2</sup> )	15,1 - 23
Fasilitas Kos	Internet/Wi-Fi

#### Foto Kamar dan Kos


#### Lokasi Kos

Minimarket Terdekat	±1.1 KM
Supermarket Terdekat	±0.9 KM
Masjid Terdekat	±1.6 KM

**Gambar 4.4** Tampilan Halaman Detail Kamar dan Indekos

4.2.4 Implementasi Halaman *Form* Pemesanan Kamar

Halaman ini menampilkan formulir pemesanan kamar. Formulir pemesanan kamar ini terdiri dari nama pemesan, nomor telepon pemesan, email pemesan, tanggal masuk, dan tanggal keluar yang harus diisi oleh pencari indekos untuk proses pemesanan. Selain itu terdapat rincian kamar yang berisi nama indekos, jenis kamar, dan harga kamar indekos yang kita sudah pilih. Jika pencari indekos mempunyai kode promo bisa dimasukkan di kolom promo. Tampilan antar muka dapat dilihat pada gambar 4.5.



Cari KosTransaksiPromoInisiasi

Pemesanan Kamar

Data Pemesan

Nama Pemesan

Id nama pemesan sesuai KTP/Passpor/SIM (tanpa tanda baca/gelar)

Nomor Telepon Pemesan

Contoh: 0812345678

Email Pemesan

Contoh: email@example.com

Tanggal Masuk

hh/bb/tttt

Tanggal Keluar

hh/bb/tttt

Kode Promo

Kode

Rincian Kamar

Nama Kos

Kos Bougenville

Jenis Kamar

Kamar Besar

Harga Kamar

Rp1,308,841/bulan

Tipe Kos


Pria

Lanjutkan Pemesanan

Gambar 4.5 Tampilan Halaman *Form* Pemesanan Kamar

## 4.2.5 Implementasi Halaman Detail Pemesanan

Halaman ini menampilkan detail pemesanan. Detail pemesanan berisi nama pemesan, nomor telepon pemesan, email pemesan, nama indekos, tipe indekos, jenis kamar, harga kamar, tanggal masuk, tanggal keluar, durasi sewa, dan total pembayaran. Tampilan antar muka dapat dilihat pada gambar 4.6.


Cari Kos   Transaksi ▾   Promo   linowsmi ▾

### Pemesanan Kamar

#### Detail Pemesanan

<b>Nama Pemesan</b> Luwandino Wismar	
<b>Nomor Telepon Pemesan</b> +62081288261196	<b>Email Pemesan</b> luwandino@hotmail.com
<b>Nama Kos</b> Kos Bougenville	<b>Tipe Kos</b> Pria
<b>Jenis Kamar</b> Kamar Besar	<b>Harga Kamar</b> Rp1,308,841/bulan
<b>Tanggal Masuk</b> 2017-05-15	<b>Tanggal Keluar</b> 2017-06-16

#### Rincian Pembayaran


Kos Bougenville	
Kamar Besar	Rp1,308,841
Durasi Sewa	2 bulan
<b>Total Pembayaran</b>	<b>Rp2,617,682</b>

Lanjut Ke Pembayaran

**Gambar 4.6** Tampilan Halaman Detail Pemesanan

#### 4.2.6 Implementasi Halaman Pembayaran

Halaman ini menampilkan detail pembayaran. Detail pembayaran berisi id transaksi, total pembayaran, dan transfer ke rekening tujuan. Tampilan antar muka dapat dilihat pada gambar 4.7.




Cari KosTransaksi▼PromoLinowsmr▼

### Pembayaran

Rincian Pembayaran

ID Transaksi	25
Total Pembayaran	Rp1,308,841

Transfer Ke



Bank BNI	3456982131
Atas Nama	PT. CariKos Cooperate


Cara Transfer melalui ATM BANK BNI

Cara Transfer melalui ATM BERSAMA

Konfirmasi Pembayaran

#### 4.2.7 Implementasi Halaman Konfirmasi Pembayaran

Halaman ini menampilkan detail pembayaran. Detail pembayaran berisi id transaksi, total pembayaran, dan rekening tujuan untuk melakukan pembayaran. Tampilan antar muka dapat dilihat pada gambar 4.8.

 CariKos  
Sistem Manajemen Properti

Cari KosTransaksi▼PromoUnowsm▼

Konfirmasi Pembayaran

Data Pembayaran

ID Transaksi

ID Transaksi yang Didapat Saat Melakukan Pemesanan

Bank Rekening Asal

Tidak Ada yang Dipilih▼

Nomor Rekening

Isi nomor rekening yang dipakai untuk pembayaran

Nama Pemegang Rekening

Nama sesuai dengan pemilik rekening

Sebesar

Rp

Total yang harus dibayar Rp1,300,041

Konfirmasi

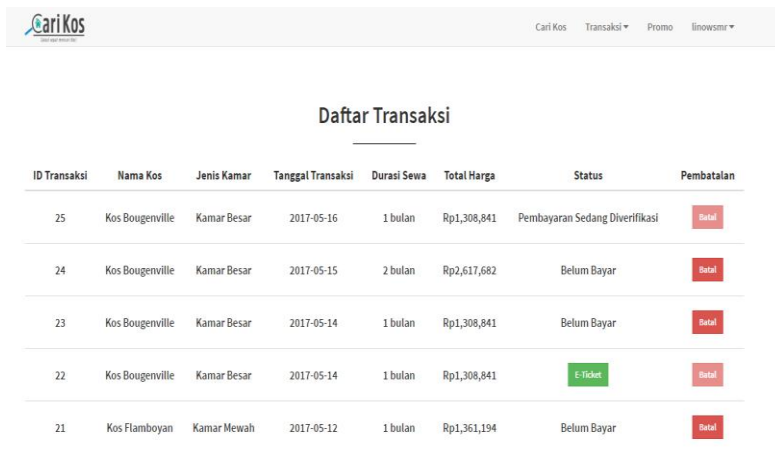
**Gambar 4.8** Tampilan Halaman Konfirmasi Pembayaran

#### 4.2.8 Implementasi Halaman Daftar Transaksi di Sisi Pencari Indekos

Halaman ini menampilkan daftar transaksi di sisi pencari indekos. Daftar transaksi berisi id transaksi, nama kos, jenis kamar, tanggal transaksi, durasi sewa, total harga, status pembayaran, dan pembatalan.

Status pembayaran berisi informasi tentang status dari pembayaran pemesanan pencari indekos. Jika pencari indekos belum melakukan pembayaran, status akan berisi belum bayar. Jika pencari indekos sudah membayar tapi belum diverifikasi admin, maka status akan berisi pembayaran sedang diverifikasi. Jika pencari indekos sudah membayar dan admin sudah verifikasi pembayaran, maka status berisi tombol *e-ticket* yang dapat melihat *e-ticket* dari pemesanan. Jika pencari indekos membatalkan pemesanan, status akan berisi batal.

Pada pembatalan berisi tombol batal yang aktif ketika pemesanan belum dibatalkan dan belum bayar. Tampilan antar muka dapat dilihat pada gambar 4.9.



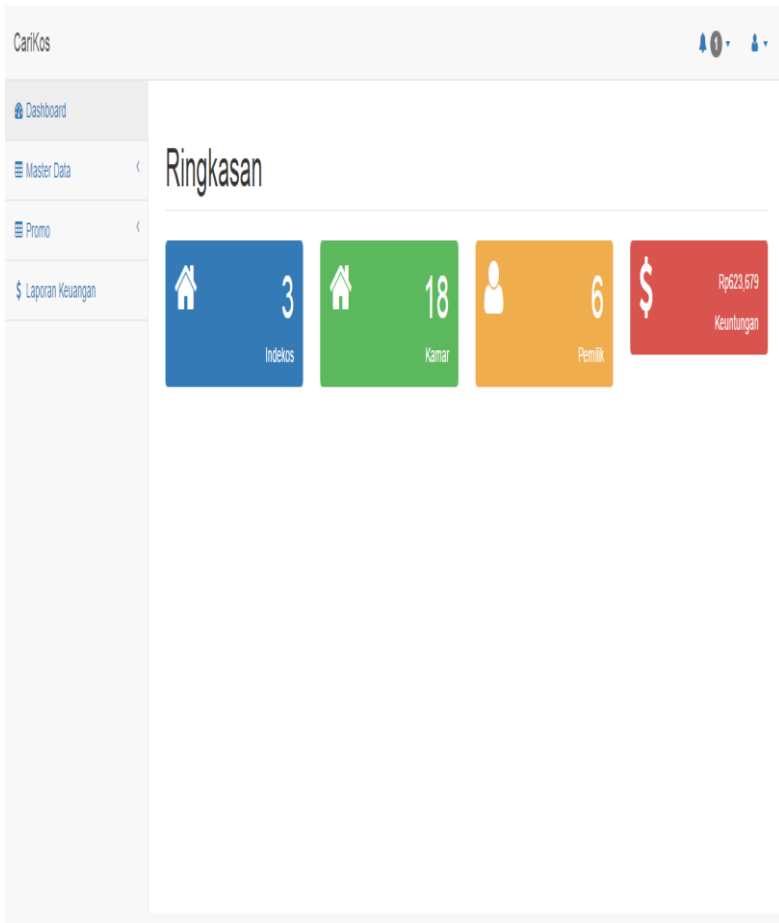
ID Transaksi	Nama Kos	Jenis Kamar	Tanggal Transaksi	Durasi Sewa	Total Harga	Status	Pembatalan
25	Kos Bougenville	Kamar Besar	2017-05-16	1 bulan	Rp1,308,841	Pembayaran Sedang Diverifikasi	Batal
24	Kos Bougenville	Kamar Besar	2017-05-15	2 bulan	Rp2,617,682	Belum Bayar	Batal
23	Kos Bougenville	Kamar Besar	2017-05-14	1 bulan	Rp1,308,841	Belum Bayar	Batal
22	Kos Bougenville	Kamar Besar	2017-05-14	1 bulan	Rp1,308,841	E-Ticket	Batal
21	Kos Flamboyen	Kamar Mewah	2017-05-12	1 bulan	Rp1,361,194	Belum Bayar	Batal

**Gambar 4.9** Tampilan Halaman Daftar Transaksi di Sisi Pencari Indekos



## 4.2.9 Implementasi Halaman Utama Admin

Halaman ini menampilkan halaman utama admin yang berisi tentang ringakasan jumlah indekos, kamar, pemilik, dan keuntungan yang didapat. Tampilan antar muka dapat dilihat pada gambar 4.10.



**Gambar 4.10 Tampilan Halaman Utama Admin**

4.2.10 Implementasi Halaman Indekos

Halaman ini menampilkan halaman daftar indekos. Daftar indekos berisi nama indekos, alamat indekos, nomor telepon indekos, jumlah kamar, dan nama pemilik indekos. Tampilan antar muka dapat dilihat pada gambar 4.11.

CaritKos

Dashboard

Master Data

Indekos

Reservasi

Transaksi

Portal

Promo

Laporan Keuangan

Indekos

Nama	Alamat	Telepon	Jumlah		Pemilik
			Kamar		
Kos Bougenville	Jalan Teknik Komputer No.10, Keputih, Kota Surabaya, Jawa Timur, Indonesia	0314514204	8		Luwandino Wismar
Kos Jasmine	Jalan Bhaskara Utara No.10, Kalisari, Kota Surabaya, Jawa Timur, Indonesia	0314895205	6		Luwandino Wismar
Kos Flamboyan	Jalan Teknik Komputer I, Keputih, Kota Surabaya, Jawa Timur, Indonesia	0314514205	4		Luwandino Wismar

Gambar 4.11 Tampilan Halaman Indekos

#### 4.2.11 Impelmentasi Halaman Kamar Indekos

Halaman ini menampilkan halaman daftar kamar indekos. Daftar kamar indekos berisi jenis kamar, jumlah kamar, dan harga kamar. Tampilan antar muka dapat dilihat pada gambar 4.12.

Jenis Kamar	Jumlah Kamar	Harga (Rp.)
Kamar Besar	8	1,250,000

Kembali

**Gambar 4.12 Tampilan Halaman Kamar Indekos**

#### 4.2.12 Implementasi Halaman Pemesanan

Halaman ini menampilkan halaman pemesanan. Halaman pemesanan berisi id pemesanan, nama pemesan, durasi pemesanan,

harga pemesanan, indekos, dan jenis kamar. Tampilan antar muka dapat dilihat pada gambar 4.13.

## Pemesanan

Show 10 entries

Search:

ID Pemesanan	Nama Pemesan	Durasi Pemesanan	Harga Pemesanan	Indekos	Kamar
139	Luwandino Wismar	1 bulan	Rp1,240,107	Kos Sakura	Kamar Superior
140	Luwandino Wismar	7 bulan	Rp1,207,060	Kos Anyelir	Kamar Deluxe
141	Luwandino Wismar	1 bulan	Rp1,207,060	Kos Anyelir	Kamar Deluxe
142	Luwandino Wismar	1 bulan	Rp1,129,667	Kos Anyelir	Kamar Deluxe
143	Luwandino Wismar	1 bulan	Rp1,257,739	Kos Matahari	Kamar Besar
144	Luwandino Wismar	1 bulan	Rp1,432,714	Kos Eidelweis	Kamar Superior
145	Luwandino Wismar	1 bulan	Rp1,432,714	Kos Eidelweis	Kamar Superior
146	Luwandino Wismar	1 bulan	Rp1,432,714	Kos Eidelweis	Kamar Superior
147	Luwandino Wismar	1 bulan	Rp1,432,714	Kos Eidelweis	Kamar Superior
148	Luwandino Wismar	1 bulan	Rp1,432,714	Kos Eidelweis	Kamar Superior

Showing 1 to 10 of 24 entries

Previous

1

2

3

Next

Gambar 4.13 Tampilan Halaman Pemesanan

4.2.13 Implementasi Halaman Transaksi

Halaman ini menampilkan halaman transaksi. Halaman transaksi berisi id transaksi, id pemesanan, tanggal pemesanan, total pembayaran, bank asal, nomor rekening, nama pemilik rekening, dan status pembayaran.

Status pembayaran berisi status dari pembayaran yang dilakukan pencari indeks. Bila pencari indeks sudah membayar, akan keluar tombol verifikasi untuk verifikasi pembayaran yang dilakukan oleh pencari indeks. Tampilan antar muka dapat dilihat pada gambar 4.14.

Carikos

Dashboard

Master Data

Indeks

Reservasi

Transaksi

Portal

Promo

Laporan Keuangan

Transaksi

ID Transaksi	ID Pemesanan	Tanggal Pembayaran	Total Pembayaran	Bank Asal	Nomor Rekening	Nama Pemilik	Status
17	125	2017-05-10 14:42:42	Rp2,612,766	Bank BRI	123456	Luwandino Wismar	Lunas
18	126	2017-05-11 22:15:54	Rp2,717,322	-	-	-	Batal
19	128	2017-05-11 22:24:37	Rp2,476,998	-	-	-	Batal
20	130	2017-05-12 12:49:59	Rp1,361,194	-	-	-	Belum bayar
21	131	2017-05-12 12:50:24	Rp1,361,194	-	-	-	Belum bayar
22	132	2017-05-14 15:11:35	Rp1,308,841	Bank BNI	5632456783	Lino	Lunas
23	133	2017-05-14 18:33:41	Rp1,308,841	-	-	-	Belum bayar
24	134	2017-05-16 08:42:07	Rp2,617,682	-	-	-	Belum bayar
25	135	2017-05-16 18:07:04	Rp1,308,841	Bank BNI	4320427453245	Lino	Verifikasi

**Gambar 4.14 Tampilan Halaman Transaksi**

#### 4.2.14 Implementasi Halaman Laporan Keuangan

Halaman ini menampilkan halaman laporan keuangan yang berfungsi untuk melihat keuntungan atau kerugian yang didapat. Halaman laporan keuangan berisi id transaksi, tanggal, harga asli, harga baru, dan total. Tampilan antar muka dapat dilihat pada gambar 4.15.

CarikOS

Dashboard

Master Data

Promo

Laporan Keuangan

### Laporan Keuangan

Bulan: Januari Tahun: 2017

ID Transaksi	Tanggal	Harga Asli (Rp)	Harga Baru (Rp)	Total (Rp)
17	2017-05-10	2,500,000	2,612,766	112,766
19	2017-05-11	2,500,000	2,476,998	-23,002
22	2017-05-14	1,250,000	1,308,841	58,841
23	2017-05-14	1,250,000	1,308,841	58,841
24	2017-05-15	2,500,000	2,617,682	117,682
25	2017-05-16	1,250,000	1,308,841	58,841
18	2017-05-11	2,600,000	2,717,322	117,322
20	2017-05-12	1,300,000	1,361,194	61,194
21	2017-05-12	1,300,000	1,361,194	61,194
Total				623,679

**Gambar 4.15** Tampilan Halaman Laporan Keuangan

### 4.3 Implementasi Penentuan Harga

Pada subbab ini akan dijelaskan tentang implementasi penentuan harga dari metode back propagation. Metode back propagation ini digunakan untuk menentukan harga yang sesuai

dengan mempertimbangkan permintaan indekos dan nilai skor dari indekos.

#### 4.3.1 Implementasi *Training Data*

*Training data* dilakukan untuk mendapatkan nilai dari setiap *weight* yang tersedia pada jaringan tiruan. Kode sumber *training data* dapat dilihat pada kode sumber 4.1.

```

1  $lr = 0.9;
2
3  $a = rand(-1,1) / 10;
4  $b = rand(-1,1) / 10;
5  $c = rand(-1,1) / 10;
6  $d = rand(-1,1) / 10;
7  $e = rand(-1,1) / 10;
8  $f = rand(-1,1) / 10;
9  $g = rand(-1,1) / 10;
10 $h = rand(-1,1) / 10;
11 $i = rand(-1,1) / 10;
12 $j = rand(-1,1) / 10;
13 $k = rand(-1,1) / 10;
14 $l = rand(-1,1) / 10;
15 $m = rand(-1,1) / 10;
16
17 $input = array
18 (
19     array(0.21,0.3),
20     array(0.85,0.42),
21     array(0.74,0.58),
22     array(0.1,0.22),
23     array(0.42,0.26),
24     array(0.14,0.58),
25     array(0.52,0.34),
26     array(0.15,0.46),
27     array(0.41,0.74),
28     array(0.1,0.66)
29 );
30
31 $output = array(0,1,1,0,0,0,0,0,1,0);
32
33 for ($x=0; $x<180; $x++) {
34     for ($y=0; $y < 10; $y++) {

```

```

35     $nh1[$y]= $a + ($b*$input[$y][1]) +
($c*$input[$y][0]);
36     $nh2[$y]= $d + ($e*$input[$y][1]) +
($f*$input[$y][0]);
37     $nh3[$y]= $g + ($h*$input[$y][1]) +
($i*$input[$y][0]);
38
39     $oh1[$y]= 1 / (1+exp($nh1[$y]*(-1)));
40     $oh2[$y]= 1 / (1+exp($nh2[$y]*(-1)));
41     $oh3[$y]= 1 / (1+exp($nh3[$y]*(-1)));
42
43     $no[$y]= $j + ($oh1[$y]*$k) + ($oh2[$y]*$l) +
($oh3[$y]*$m);
44
45     $out[$y] = 1 / (1+exp($no[$y]*(-1)));
46
47     $mse[$y] = pow(($out[$y]-$output[$y]),2);
48
49     $err[$y] = ($output[$y]-$out[$y])*out[$y]*(1-
$out[$y]);
50
51     $uj = $j + $lr*$err[$y];
52     $uk = $k + $lr*$err[$y]*$oh1[$y];
53     $ul = $l + $lr*$err[$y]*$oh2[$y];
54     $um = $m + $lr*$err[$y]*$oh3[$y];
55
56     $enh1 = $err[$y]*$k;
57     $enh2 = $err[$y]*$l;
58     $enh3 = $err[$y]*$m;
59
60     $eh1 = $enh1*$oh1[$y]*(1-$oh1[$y]);
61     $eh2 = $enh2*$oh2[$y]*(1-$oh2[$y]);
62     $eh3 = $enh3*$oh3[$y]*(1-$oh3[$y]);
63
64     $ua = $a + $lr*$eh1;
65     $ub = $b + $lr*$eh1*$input[$y][1];
66     $uc = $c + $lr*$eh1*$input[$y][0];
67     $ud = $d + $lr*$eh2;
68     $ue = $e + $lr*$eh2*$input[$y][1];
69     $uf = $f + $lr*$eh2*$input[$y][0];
70     $ug = $g + $lr*$eh3;
71     $uh = $h + $lr*$eh3*$input[$y][1];
72     $ui = $i + $lr*$eh3*$input[$y][0];
73
74     $a = $ua;
75     $b = $ub;
76     $c = $uc;
77     $d = $ud;

```



```

78         $e = $ue;
79         $f = $uf;
80         $g = $ug;
81         $h = $uh;
82         $i = $ui;
83         $j = $uj;
84         $k = $uk;
85         $l = $ul;
86         $m = $um;
87     }
88     $hasilmse[$x] = (array_sum($mse))/10;
89 }
$totalmse = (array_sum($hasilmse))/$x;

```

**Kode Sumber 4.1 Training Data**

### 4.3.2 Implementasi Testing Data

Penentuan harga dapat dilakukan setelah melakukan *training* data. Dari setiap *weight* yang didapatkan dari *training* data dipakai ketika melakukan *testing* data. Kode sumber *testing* data dapat dilihat di kode sumber 4.2.

```

1  $bulan = date("m");
2
3  $a = 1.2817164322232;
4  $b = -2.0161081024844;
5  $c = -2.4820055708634;
6  $d = 0.39234709422736;
7  $e = -1.4500143409141;
8  $f = -1.8379153691184;
9  $g = 3.2170084420988;
10 $h = -3.7499228156667;
11 $i = -3.788658534876;
12 $j = 3.5249716409331;
13 $k = -3.2197327590555;
14 $l = -2.0392787777114;
15 $m = -5.9333871155382;
16
17 if($bulan==1){
18     $nilaiBulan = 8;
19 }
20 if($bulan==11 || $bulan == 2){
21     $nilaiBulan = 2;
22 }

```

```

23  if($bulan==3 || $bulan == 4){
24      $nilaiBulan = 4;
25  }
26  if($bulan == 5){
27      $nilaiBulan = 15;
28  }
29  if($bulan==6){
30      $nilaiBulan = 25;
31  }
32  if($bulan==7){
33      $nilaiBulan = 32;
34  }
35  if($bulan==8){
36      $nilaiBulan = 35;
37  }
38  if($bulan==9){
39      $nilaiBulan = 29;
40  }
41  if($bulan==10){
42      $nilaiBulan = 26;
43  }
44  if($bulan==12){
45      $nilaiBulan = 3;
46  }
47
48  $nilaiBln = (((0.8*$nilaiBulan)-(0.8*2))/(35-2))+0.1;
    $nilaiSkor = (((0.8*$skor)-(0.8*0))/(100-0))+0.1;
49
50  $nh1 = $a + $nilaiSkor*$b + $nilaiBln*$c;
51  $nh2 = $d + $nilaiSkor*$d + $nilaiBln*$f;
52  $nh3 = $g + $h*$nilaiSkor + $i*$nilaiBln;
53
54  $oh1 = 1 / (1+exp($nh1*(-1)));
55  $oh2 = 1 / (1+exp($nh2*(-1)));
56  $oh3 = 1 / (1+exp($nh3*(-1)));
57
58  $no= $j + ($oh1*$k) + ($oh2*$l) + ($oh3*$m);
59  $out = 1 / (1+exp($no*(-1)));
60
61  $hargaLama = $row->hargaKamar;
62  $max = $hargaLama + (0.05*$hargaLama);
63  $min = $hargaLama - (0.025*$hargaLama);
64
65  $hargaBaru = ((($max-$min)*($out-0.1))/0.8)+$min;
66

```

**Kode Sumber 4.2 Testing Data**

## 4.4 Implementasi Alur Proses Aplikasi

Pada subbab ini akan dijelaskan tentang implementasi alur proses yang terjadi di aplikasi dalam sebuah kode. Proses yang dibahas adalah proses-proses yang terjadi dalam sistem sesuai dengan kebutuhan fungsional.

### 4.4.1 Implementasi Alur Proses Pencarian

Proses pencarian ini digunakan untuk mencari kamar indeks yang sesuai dengan keinginan pencari indeks. Implementasi pencarian indeks ini menggunakan fungsi `index()` untuk menampilkan hasil pencarian kamar indeks yang sesuai dengan keinginan pencari indeks. *Input* dari implementasi ini adalah nama kota, *range* harga, tipe indeks, fasilitas kamar, fasilitas indeks, dan jurusan yang ingin dicari. Sedangkan *output* dari implementasi ini adalah daftar indeks yang sesuai kriteria yang sudah dimasukkan. Kode program dapat dilihat pada Kode Sumber 4.3.

1	<code>public function index()</code>
2	<code>{</code>
3	<code>    \$kota = strtolower(\$this-&gt;input-&gt;get('kota'));</code>
4	<code>    \$harga = \$this-&gt;input-&gt;get('harga');</code>
5	
6	<code>    if(\$harga == 1){</code>
7	<code>        \$minHarga = 0;</code>
8	<code>        \$maxHarga = 500000;</code>
9	<code>    }</code>
10	<code>    else if(\$harga == 2){</code>
11	<code>        \$minHarga = 500001;</code>
12	<code>        \$maxHarga = 1000000;</code>
13	<code>    }</code>
14	<code>    else if(\$harga == 3){</code>
15	<code>        \$minHarga = 1000001;</code>
16	<code>        \$maxHarga = 1500000;</code>
17	<code>    }</code>
18	<code>    else if(\$harga == 4){</code>
19	<code>        \$minHarga = 1500001;</code>
20	<code>        \$maxHarga = 9999999;</code>
21	<code>    }</code>

```

22
23     $tipe = $this->input->get('tipe');
24     $kos = $this->input->get('fasilitaskos');
25     $kamar = $this->input->get('fasilitaskamar');
26     $jurusanDipilih = $this->input->get('jurusan');
27     $fasilitaskos = implode(",", $kos);
28     $fasilitaskamar = implode(",", $kamar);
29
30     if($jurusanDipilih != ""){
31         $jurusan = $this->model_pencarian-
            >ambil_jurusan($jurusanDipilih);
32
33         foreach ($jurusan as $row) {
34             $idJurusan = $row->idJurusan;
35         }
36
37         $data['hasil'] = $this->model_pencarian-
            >pencarian_jurusan($kota, $minHarga, $maxHarga,
            $tipe, $fasilitaskos, $fasilitaskamar, $idJurusan);
            $data['idJurusan'] = $jurusanDipilih;
38     }
39     else{
40         $data['hasil'] = $this->model_pencarian-
            >pencarian($kota, $minHarga, $maxHarga, $tipe,
            $fasilitaskos, $fasilitaskamar);
            $data['idJurusan'] = 0;
42     }
43
44     $data['kamarTerpakai'] = $this->model_kamar-
45     >terpakai();
46
47     $this->load->view('template/header');
48     $this->load->view('hasil_pencarian', $data);
49     $this->load->view('template/footer');
50 }

```

**Kode Sumber 4.3 Fungsi Proses Pencarian**

#### 4.4.2 Implementasi Alur Proses Pemesanan

Proses pemesanan ini digunakan untuk memesan kamar indekos. Implementasi pemesanan ini menggunakan fungsi pesan() untuk menampilkan formulir pengisian pemesanan kamar indekos. *Input* dari implementasi ini adalah username dari pencari indekos, nama pencari indekos, telepon pencari indekos, email pencari

indekos, tanggal masuk, tanggal keluar, id indekos, id kamar, harga kamar, dan total harga pemesanan. *Output* dari implementasi ini adalah melakukan pemesanan dengan cara mencatat *input* ke dalam *database*. Kode program dapat dilihat pada Kode Sumber 4.4.

```

1 public function pesan(){
2     if(!empty($this->session->userdata('logged_in_akun'))){
3         $session_data = $this->session-
>userdata('logged_in_akun');
4         $dataAkun['username'] = $session_data['username'];
5
6         $username = $this->input->post('username');
7         $data['nama'] = $this->input->post('nama');
8         $kodeTelepon = "+62";
9         $data['telepon'] = $kodeTelepon.$this->input-
>post('telepon');
10        $data['email'] = $this->input->post('email');
11        $data['masuk'] = $this->input->post('masuk');
12        $data['keluar'] = $this->input->post('keluar');
13
14        $masuk = date_create($data['masuk']);
15        $keluar = date_create($data['keluar']);
16
17        $diff = date_diff($masuk, $keluar);
18
19        $tahun = $diff->format("%y");
20        $bulan = $diff->format("%m");
21        $hari = $diff->format("%d");
22
23        $data['durasi'] = 0;
24        if($tahun > 0)
25            $data['durasi'] = $data['durasi'] + $tahun*12;
26        if($bulan > 0)
27            $data['durasi'] = $data['durasi'] + $bulan;
28        if($hari > 0)
29            $data['durasi'] = $data['durasi'] + 1;
30        if($data['durasi'] < 1)
31            echo "Salah Input Tanggal";
32
33        $idKamar = $this->input->post('kamar');
34        $idKos = $this->input->post('kos');
35        $data['harga'] = $this->input->post('harga');
36        $data['totalPembayaran'] =
    $data['durasi']*$data['harga'];
37        if($data['idPemesanan'] != "Gagal"){

```

```

38         $this->load->view('template/header_akun',
$dataAkun);
39         $this->load->view('detail_pemesanan', $data);
40         $this->load->view('template/footer');
41     }
42     else
43         echo "Gagal";
44 }
45 else
46     redirect('pemilik/masuk');
47 }

```

#### 4.4.3 Implementasi Alur Proses Pembatalan Pemesanan

Proses pemesanan ini digunakan untuk melakukan pembatalan pemesanan. Implementasi pembatalan pemesanan ini menggunakan fungsi pembatalan() untuk menampilkan formulir pengisian pemesanan kamar indekos. *Input* dari implementasi ini adalah id transaksi yang ingin dibatalkan. *Output* dari implementasi ini adalah merubah status menjadi 3 yang merupakan status batal dalam *database*. Kode program dapat dilihat pada Kode Sumber 4.5.

```

1  public function pembatalan(){
2      if(!empty($this->session->userdata('logged_in_akun'))){
3          $session_data = $this->session-
>userdata('logged_in_akun');
4          $dataAkun['username'] = $session_data['username'];
5
6          $idTransaksi = $this->input->get('transaksi');
7          $data['akun'] = $this->model_transaksi-
>cek_transaksi($idTransaksi);
8          foreach($data['akun'] as $row){
9              $akun = $row->username;
10         }
11         if($akun == $dataAkun['username']){
12             $status = 3;
13             $batal = $this->model_transaksi-
>pembatalan_transaksi($idTransaksi, $status);
14             redirect('transaksi/daftar');
15         }

```

```

16         else{
17             echo "Bukan Transaksi Anda";
18         }
19     }
20     else
21         echo "Anda Harus Login Terlebih Dahulu";
22 }

```

**Kode Sumber 4.5 Fungsi Pembatalan Pemesanan**

#### 4.4.4 Implementasi Alur Proses Pembayaran

Proses pembayaran ini digunakan untuk melakukan proses pembayaran pemesanan. Implementasi pembayaran ini menggunakan fungsi `konfBayar()` untuk menampilkan formulir pengisian konfirmasi pembayaran yang sudah dilakukan pencari indekos. *Input* dari implementasi ini adalah id transaksi, nomor rekening pencari indekos, nama rekening, dan bank asal. Bila pencari indekos tidak mengisi formulir, maka konfirmasi pembayaran tidak dapat terbaca oleh sistem. Kode program dapat dilihat pada Kode Sumber 4.6.

```

1  public function konfBayar(){
2      $idTransaksi = $this->input->post('idTransaksi');
3      $norek = $this->input->post('norek');
4      $namarek = $this->input->post('namarek');
5      $bank = $this->input->post('bank');
6      $status = 1;
7
8      $cek = $this->model_transaksi-
>verifikasi($idTransaksi,$norek,$namarek,$bank,$status);
9      if($cek != 'Gagal'){
10         redirect('home');
11     }
12     else{
13         echo 'Gagal';
14     }
15 }

```

**Kode Sumber 4.6 Fungsi Pembayaran**

#### 4.4.5 Implementasi Alur Proses Mendapatkan *E-Ticket*

Proses mendapatkan *e-ticket* ini digunakan untuk mendapatkan *e-ticket* dari pemesanan yang sudah dibayarkan. Implementasi mendapatkan *e-ticket* ini menggunakan fungsi `eticket()` untuk memanggil fungsi yang membuat *e-ticket* yang menggunakan id transaksi sebagai *input*. *Output* yang didapatkan merupakan *e-ticket* yang berisi id transaksi, nama pemesan, nama indekos, jenis kamar, tanggal masuk, tanggal keluar, alamat indekos, dan foto indekos. *E-ticket* ini digunakan sebagai tanda bahwa pencari indekos sudah melakukan pembayaran dan admin aplikasi CariKos sudah melakukan konfirmasi permabayaran pencari indekos. Kode program dapat dilihat pada Kode Sumber 4.7.

```

1 public function eticket(){
2     if(!empty($this->session->userdata('logged_in_akun'))){
3         $session_data = $this->session-
4 >userdata('logged_in_akun');
5         $dataAkun['username'] = $session_data['username'];
6
7         $idTransaksi = $this->input->get('transaksi');
8         $data['akun'] = $this->model_transaksi-
9 >cek_transaksi($idTransaksi);
10        foreach($data['akun'] as $row){
11            $akun = $row->username;
12
13            if($akun == $dataAkun['username']){
14                $data['eticket'] = $this->model_transaksi-
15 >eticket($idTransaksi);
16
17                $this->load->view('eticket',$data);
18            }
19            else{
20                echo "Bukan Transaksi Anda";
21            }
22        }
23    }
24    else
25        echo "Anda Harus Login Terlebih Dahulu";
26 }

```

**Kode Sumber 4.7 Fungsi Mendapatkan E-Ticket**



Proses membuat *e-ticket* ini digunakan untuk membuat *e-ticket* dari pemesanan yang sudah dibayarkan. Implementasi membuat *e-ticket* ini menggunakan *library* TCPDF. Dengan *library* TCPDF ini, dapat mengubah halaman HTML menjadi halaman PDF. Kode program dapat dilihat pada Kode Sumber 4.8.

```

1  require_once('assets/TCPDF/examples/tcpdf_include.php');
2  require_once('assets/TCPDF/tcpdf.php');
3
4  foreach ($ticket as $row) {
5      $idTransaksi = $row->idTransaksi;
6      $tanggalMasuk = $row->tanggalMasuk;
7      $tanggalKeluar = $row->tanggalKeluar;
8      $jenisKamar = $row->jenisKamar;
9      $namaKos = $row->namaKos;
10     $alamatKos = $row->alamatKos;
11     $teleponKos = $row->teleponKos;
12     $namaFileKamar = $row->namaFileKamar;
13     $namaFile = $row->namaFile;
14     $namaAkun = $row->namaAkun;
15 }
16
17 $pdf = new
    TCPDF(PDF_PAGE_ORIENTATION,PDF_UNIT,PDF_PAGE_FORMAT, true,
        'UTF-8',false);
18
19 $pdf->SetCreator(PDF_CREATOR);
20 $pdf->SetAuthor('CariKos');
21 $pdf->SetTitle('e-ticket');
22
23 $pdf->SetHeaderData(PDF_HEADER_LOGO, PDF_HEADER_LOGO_WIDTH,
    '',PDF_HEADER_STRING, array(0,64,255), array(0,64,128));
24 $pdf->SetFooterData(array(0,64,0), array(0,64,128));
25
26 $pdf->setHeaderFont(Array(PDF_FONT_NAME_MAIN, '',
    PDF_FONT_SIZE_MAIN));
27 $pdf->setFooterFont(Array(PDF_FONT_NAME_DATA, '',
    PDF_FONT_SIZE_DATA));
28
29 $pdf->SetDefaultMonospacedFont(PDF_FONT_MONOSPACED);
30
31 $pdf->SetMargins(PDF_MARGIN_LEFT, PDF_MARGIN_TOP,
    PDF_MARGIN_RIGHT);
32 $pdf->SetHeaderMargin(PDF_MARGIN_HEADER);

```

```

33 $pdf->SetFooterMargin(PDF_MARGIN_FOOTER);
34
35 // set auto page breaks
36 $pdf->SetAutoPageBreak(TRUE, PDF_MARGIN_BOTTOM);
37
38 // set image scale factor
39 $pdf->setImageScale(PDF_IMAGE_SCALE_RATIO);
40
41 $pdf->setFontSubsetting(true);
42
43 $pdf->SetFont('times', '', 8, '', true);
44 $pdf->setCellHeightRatio(0.5);
45
46 $pdf->AddPage();
47
48 // set text shadow effect
49 $pdf->setTextShadow(array('enabled'=>true, 'depth_w'=>0.2,
    'depth_h'=>0.2, 'color'=>array(196,196,196), 'opacity'=>1,
    'blend_mode'=>'Normal'));
50
51 // Set some content to print
52 $gambar='
53 
55 '
56 $detail='
57 <p style="font-size:14px;"><i>Nama</i></p>
58 <h2>'. $namaAkun. '</h2>
59 </br>
60 <p style="font-size:14px;"><i>Jenis Kamar</i></p>
61 <h2>'. $jenisKamar. '</h2>
62 </br>
63 <p style="font-size:14px;"><i>Tanggal Masuk</i></p>
64 <h2>'. $tanggalMasuk. '</h2>
65 </br>
66 <p style="font-size:14px;"><i>Tanggal Keluar</i></p>
67 <h2>'. $tanggalKeluar. '</h2>
68 </br>
69 '
70
71 $head = '
72 <h1 style="text-align: center; font-size: 30px;">E-
73 TICKET</h1>
74 <p style="text-align: center; font-size: 30px;">Pemesanan
75 Indekos</p>
76 <hr>
  
```

```

77 <h1>ID TRANSAKSI : '.$idTransaksi.'</h1>
78 </br>
79 <h1>Indekos '.$namaKos.'</h1>
80 </br>
81 <p style="font-style:italic; font-
size:12px;">'.$alamatKos.'</p>
82 <p style="font-size:12px;">'.$teleponKos.'</p>
83 <hr>
84 ';
85
86 $head1 = '
87 <h1>Indekos '.$namaKos.'</h1>
88 <p style="font-style:italic; font-
size:12px;">'.$alamatKos.'</p>
89 <p style="font-size:12px;">'.$teleponKos.'</p>
90 ';
91
92 $tglMasuk = '
93 <h3>Tanggal Masuk</h3>
94 <h3>'.$tanggalMasuk.'</h3>
95 ';
96
97 $tglKeluar = '
98 <h3>Tanggal Keluar</h3>
99 <h3>'.$tanggalKeluar.'</h3>
100 ';
101
102 // Print text using writeHTMLCell()
103 $pdf->writeHTMLCell(0, 0, '', '30', $head, 0, 1, 0, true,
'', true);
104 $pdf->writeHTMLCell(0, 0, '', '85', $detail, 0, 1, 0, true,
'', true);
105 $pdf->writeHTMLCell(0, 0, '110', '85', $gambar, 0, 1, 0,
true, '', true);
106 $pdf->Output('example_001.pdf', 'I');

```

**Kode Sumber 4.8 Fungsi Membuat E-Ticket**

#### 4.4.6 Implementasi Alur Proses Melihat Kamar Indekos

Proses melihat kamar indekos ini digunakan untuk melihat kamar indekos yang terdaftar di CariKos. Implementasi melihat kamar indekos ini menggunakan fungsi lihatKamar() untuk menampilkan kamar indekos yang terdaftar di CariKos. *Input* dari implementasi adalah id indekos dan *output* yang dihasilkan adalah menampilkan tabel kamar indekos yang terdaftar di CariKos. Tabel

kamar indekos berisi informasi tentang kamar indekos dari indekos yang dipilih. Informasi tersebut berisi jenis kamar, jumlah kamar yang tersedia, dan harga dasar dari kamar indekos tersebut, Kode program dapat dilihat pada Kode Sumber 4.9.

```

1 public function lihatKamar(){
2     if(!empty($this->session->userdata('logged_in_admin'))){
3         $session_data = $this->session-
>userdata('logged_in_admin');
4         $nama['username'] = $session_data['username'];
5         $nama['notifTransaksi'] = $this->model_transaksi-
>notifTransaksi();
6
7         $id = $this->input->get('kos');
8         $data['id'] = $id;
9         $data['namaKos'] = $this->model_kos->namaKos($id);
10        $data['kamar'] = $this->model_kamar->lihatKamar($id);
11
12        $this->load->view('admin/admheader',$nama);
13        $this->load->view('admin/kamar', $data);
14    }
15    else{
16        redirect('admin');
17    }
18 }

```

**Kode Sumber 4.9 Fungsi Lihat Kamar Indekos**

#### 4.4.7 Implementasi Alur Proses Melihat Indekos

Proses melihat indekos ini menggunakan fungsi lihatKos() untuk menampilkan indekos yang terdaftar di CariKos. Kode program dapat dilihat pada Kode Sumber 4.10.

```

1 public function lihatKos(){
2     $nama['username'] = $session_data['username'];
3     $nama['notifTransaksi'] = $this->model_transaksi-
>notifTransaksi();
4     $data['kos'] = $this->model_kos->lihatKos();
5     $id = $this->input->get('kos');
6     $data['id'] = $id;
7     $this->load->view('admin/admheader',$nama);
8     $this->load->view('admin/indekos', $data);
9 }

```

9	else{
10	redirect('admin');
11	}

**Kode Sumber 4.10 Fungsi Melihat Indekos**

#### 4.4.8 Implementasi Alur Proses Melihat Transaksi

Proses melihat transaksi ini digunakan untuk melihat daftar transaksi yang terjadi di aplikasi CariKos. Implementasi pembayaran ini menggunakan fungsi trans() yang berfungsi untuk menampilkan daftar transaksi. Kode program dapat dilihat pada Kode Sumber 4.6.

1	public function trans(){
2	if(!empty(\$this->session->userdata('logged_in_admin'))){
3	\$session_data = \$this->session->
	userdata('logged_in_admin');
4	\$nama['username'] = \$session_data['username'];
5	\$nama['notifTransaksi'] = \$this->model_transaksi->
	notifTransaksi();
6	\$data['transaksi'] = \$this->model_transaksi->
	data_transaksi();
7	
8	\$this->load->view('admin/admheader',\$nama);
9	\$this->load->view('admin/transaksi', \$data);
10	}
11	else{
12	redirect('admin');
13	}
14	}

**Kode Sumber 4.11 Fungsi Melihat Transaksi**

#### 4.4.9 Implementasi Alur Proses Verifikasi Pembayaran

Proses verifikasi pembayaran ini digunakan untuk melakukan proses verifikasi pembayaran. Implementasi verifikasi pembayaran ini menggunakan fungsi verTrans() untuk melakukan verifikasi transaksi yang sudah dibayar pencari Indekos. *Input* pada implementasi ini adalah id transaksi dan status. *Output* yang

dihasilkan adalah perubahan status menjadi sudah diverifikasi pada *database*. Kode program dapat dilihat pada Kode Sumber 4.12.

1	public function verTrans(){
2	\$status = 2;
3	\$idTransaksi = \$this->input->post('transaksi');
4	\$update = \$this->model_transaksi-
5	>updateVerifikasi(\$idTransaksi,\$status);
6	if(\$cek != 'Gagal'){
7	redirect('admin/trans');
8	}
9	else{
10	echo 'gagal';
10	}

**Kode Sumber 4.12 Fungsi Verifikasi Pembayaran**

#### 4.4.10 Implementasi Alur Proses Melihat Laporan Keuangan

Proses melihat laporan menggunakan fungsi `lapKeu()` untuk menampilkan laporan keuangan `CariKos`. Kode program dapat dilihat pada Kode Sumber 4.13.

1	public function lapkeu(){
2	\$bulan = 0;
3	\$tahun = 0;
4	\$nama['username'] = \$session_data['username'];
5	\$nama['notifTransaksi'] = \$this->model_transaksi-
6	>notifTransaksi();
7	\$bulan = \$this->input->get('bulan');
8	\$tahun = \$this->input->get('tahun');
9	\$data['transaksi'] = \$this->model_transaksi-
10	>laporanKeuangan(\$bulan,\$tahun);
11	\$this->load->view('admin/admheader',\$nama);
12	\$this->load->view('admin/laporan',\$data);
13	}
14	else{
15	redirect('admin');
15	}

**Kode Sumber 4.13 Fungsi Melihat Laporan Keuangan**

#### 4.4.11 Implementasi Alur Proses Melihat Pemesanan

Proses melihat pemesanan ini digunakan untuk melakukan proses melihat pemesanan. Implementasi melihat pemesanan ini menggunakan fungsi reservasi() untuk menampilkan pemesanan yang terjadi di CariKos. Kode program dapat dilihat pada Kode Sumber 4.6.

1	public function reservasi(){
2	if(!empty(\$this->session->userdata('logged_in_admin'))){
3	\$session_data = \$this->session->
	userdata('logged_in_admin');
4	\$nama['username'] = \$session_data['username'];
5	\$nama['notifTransaksi'] = \$this->model_transaksi->
	notifTransaksi();
6	\$data['pemesanan'] = \$this->model_pemesanan->
	daftar_pemesanan();
7	
8	\$this->load->view('admin/admheader',\$nama);
9	\$this->load->view('admin/reservasi',\$data);
10	}
11	else{
12	redirect('admin');
13	}
14	}

**Kode Sumber 4.14 Fungsi Melihat Pemesanan**

*[Halaman ini sengaja dikosongkan]*



## **BAB V**

### **UJI COBA DAN EVALUASI**

Bab ini akan dijelaskan pengujian dan evaluasi dari aplikasi CariKos yang berbasis *web*. Pengujian yang dilakukan adalah pengujian terhadap metode back propagation sebagai penentuan harga dan pengujian terhadap kebutuhan fungsional. Pengujian fungsional mengacu pada kasus penggunaan pada bab tiga. Hasil evaluasi menjabarkan tentang rangkuman hasil pengujian pada bagian akhir bab ini.

#### **5.1 Lingkungan Pengujian**

Lingkungan pengujian aplikasi CariKos berbasis *web* dan metode back propagation dilakukan dengan spesifikasi sebagai berikut:

Jenis	: Komputer
Tipe	: Lenovo 10093
Prosesor	: Intel® Core™ i3-3240 CPU (3.40 GHz)
Memori/RAM	: 8 GB

#### **5.2 Skenario Uji Coba**

Pada bagian ini akan dijelaskan tentang skenario yang akan dilakukan. Skenario pengujian terdapat empat uji coba, yaitu uji coba back propagation, uji coba fungsionalitas, uji coba tingkat kepuasan partisipan kepada aplikasi CariKos serta pengukuran keuntungan/kerugian pada aplikasi CariKos di satu periode, uji coba penentuan harga. Untuk pengujian back propagation dilakukan percobaan *training* data sehingga mendapatkan *learning rate*, *hidden layer*, dan iterasi yang sesuai dan menghasilkan MSE yang kecil. Lalu untuk pengujian fungsional dilakukan oleh pengembang dan partisipan dengan menjalankan aplikasi dari sisi admin dan pencari indeks. Untuk pengujian tingkat kepuasan dan pengukuran keuntungan dilakukan oleh partisipan untuk

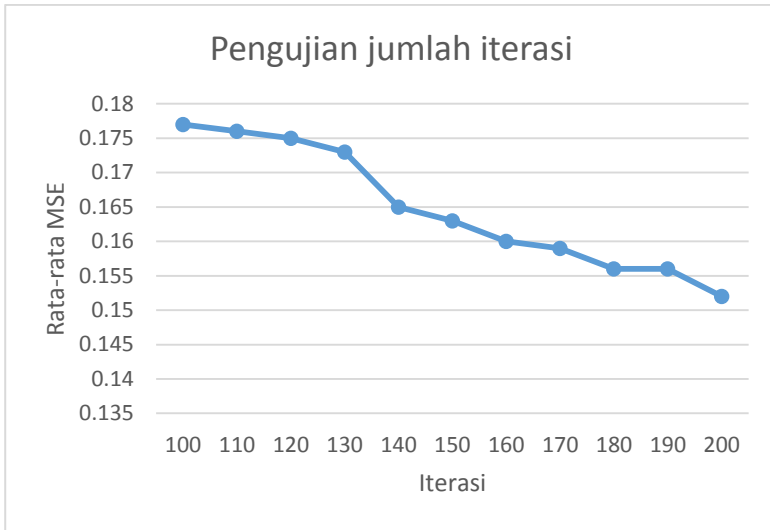
melakukan pemesanan indeks dalam satu periode yang nantinya dapat terlihat keuntungan/kerugian yang dialami aplikasi CariKos dan tingkat kepuasan dalam penggunaan aplikasi CariKos. Partisipan yang akan menguji aplikasi CariKos berjumlah lima orang. Uji coba penentuan harga untuk melihat perbedaan harga dari harga awal dengan hasil penentuan harga dari metode back propagation.

### 5.2.1 Skenario Uji Coba *Back Propagation*

Uji coba back propagation dilakukan ketika pelatihan data agar mendapatkan *learning rate*, *hidden layer*, dan iterasi yang sesuai, sehingga mendapatkan MSE yang kecil. Hal yang pertama yang dilakukan adalah mencari iterasi yang sesuai. Uji coba untuk mendapatkan iterasi yang sesuai dapat dilihat pada Tabel 5.1 dan Gambar 5.1.

**Tabel 5.1 Hasil Uji Coba Jumlah Iterasi**

Jumlah Iterasi	Nilai MSE					Rata-rata MSE
	Percobaan ke-i					
	1	2	3	4	5	
100	<b>0.176</b>	<b>0.175</b>	<b>0.180</b>	<b>0.176</b>	<b>0.178</b>	0.177
110	<b>0.173</b>	<b>0.177</b>	<b>0.176</b>	<b>0.180</b>	<b>0.176</b>	0.176
120	<b>0.177</b>	<b>0.171</b>	<b>0.180</b>	<b>0.177</b>	<b>0.172</b>	0.175
130	<b>0.172</b>	<b>0.168</b>	<b>0.180</b>	<b>0.178</b>	<b>0.167</b>	0.173
140	<b>0.156</b>	<b>0.168</b>	<b>0.164</b>	<b>0.173</b>	<b>0.164</b>	0.165
150	<b>0.168</b>	<b>0.175</b>	<b>0.152</b>	<b>0.169</b>	<b>0.152</b>	0.163
160	<b>0.143</b>	<b>0.151</b>	<b>0.176</b>	<b>0.162</b>	<b>0.166</b>	0.160
170	<b>0.167</b>	<b>0.154</b>	<b>0.147</b>	<b>0.161</b>	<b>0.167</b>	0.159
180	<b>0.164</b>	<b>0.156</b>	<b>0.152</b>	<b>0.165</b>	<b>0.142</b>	0.156
190	<b>0.162</b>	<b>0.144</b>	<b>0.148</b>	<b>0.178</b>	<b>0.146</b>	0.156
200	<b>0.151</b>	<b>0.149</b>	<b>0.149</b>	<b>0.159</b>	<b>0.152</b>	0.152



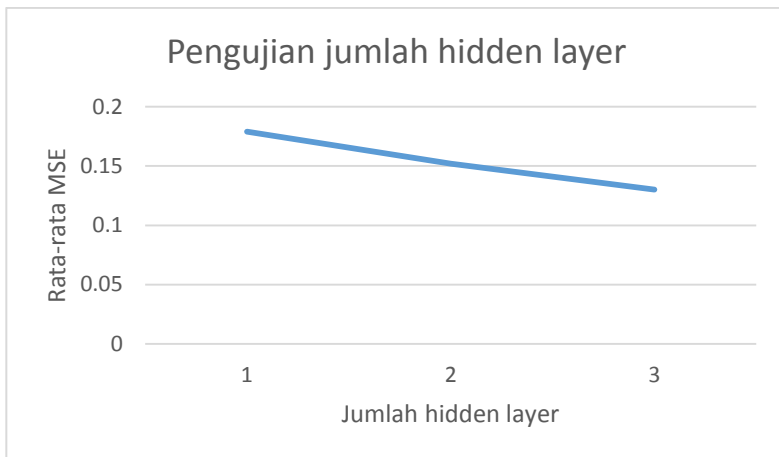
**Gambar 5.1 Grafik Pengujian Iterasi**

Dapat dilihat pada Tabel 5.1 dan Gambar 5.1 mulai iterasi ke- 180 nilai rata-rata MSE sudah stabil dan tidak terjadi penurunan atau kenaikan yang signifikan.

Pengujian yang kedua adalah pengujian jumlah *hidden layer*. Jumlah *hidden layer* yang akan diujikan berada di interval jumlah *input* dan *output*. Hasil percobaannya dapat dilihat pada Tabel 5.2 dan Gambar 5.2.

**Tabel 5.2 Hasil Pengujian Hidden Layer**

Hidden layer	Nilai MSE					Rata-rata MSE
	Percobaan ke-i					
	1	2	3	4	5	
1	<b>0.172</b>	<b>0.186</b>	<b>0.168</b>	<b>0.180</b>	<b>0.189</b>	0.179
2	<b>0.151</b>	<b>0.149</b>	<b>0.149</b>	<b>0.159</b>	<b>0.152</b>	0.152
3	<b>0.127</b>	<b>0.137</b>	<b>0.131</b>	<b>0.127</b>	<b>0.128</b>	0.130



**Gambar 5.2 Grafik Pengujian Hidden Layer**

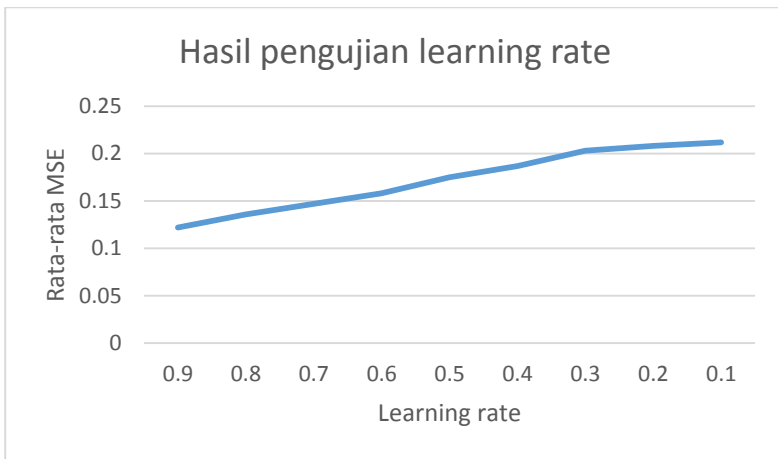
Dapat dilihat pada Tabel 5.2 dan Gambar 5.2 dapat dilihat yang paling optimal adalah disaat *hidden layer* berjumlah 3 karena nilai MSE lebih kecil dibandingkan dengan nilai MSE dari jumlah *hidden layer* yang lain sehingga untuk *training* data menggunakan *hidden layer* yang berjumlah 3.

Pengujian *learning rate* sama halnya dengan pengujian pertama dan kedua yang mana bertujuan untuk mendapatkan nilai *learning rate* yang optimal. Hasil pengujian dapat dilihat pada Tabel 5.3 dan Gambar 5.3.

**Tabel 5.3 Hasil Pengujian Learning Rate**

Learning Rate	Nilai MSE					Rata-rata MSE
	Percobaan ke-i					
	1	2	3	4	5	
0.9	<b>0.121</b>	<b>0.125</b>	<b>0.131</b>	<b>0.120</b>	<b>0.112</b>	0.122
0.8	<b>0.134</b>	<b>0.136</b>	<b>0.136</b>	<b>0.139</b>	<b>0.137</b>	0.136
0.7	<b>0.151</b>	<b>0.145</b>	<b>0.146</b>	<b>0.150</b>	<b>0.142</b>	0.147

0.6	<b>0.164</b>	<b>0.158</b>	<b>0.154</b>	<b>0.160</b>	<b>0.152</b>	0.158
0.5	<b>0.171</b>	<b>0.178</b>	<b>0.172</b>	<b>0.177</b>	<b>0.178</b>	0.175
0.4	<b>0.179</b>	<b>0.196</b>	<b>0.188</b>	<b>0.187</b>	<b>0.183</b>	0.187
0.3	<b>0.207</b>	<b>0.203</b>	<b>0.203</b>	<b>0.206</b>	<b>0.199</b>	0.203
0.2	<b>0.206</b>	<b>0.209</b>	<b>0.209</b>	<b>0.209</b>	<b>0.208</b>	0.208
0.1	<b>0.212</b>	<b>0.211</b>	<b>0.212</b>	<b>0.212</b>	<b>0.212</b>	0.212



**Gambar 5.3 Grafik Hasil Pengujian Learning Rate**

Dapat dilihat pada Tabel 5.3 dan Gambar 5.3 bahwa nilai MSE terkecil adalah ketika *learning rate* bernilai 0.9. Maka dari itu nilai *learning rate* yang akan dipakai pada saat *training* data adalah 0.9.

### 5.2.2 Skenario Uji Coba Fungsionalitas

Pada skenario uji coba yang kedua ini merupakan uji coba terhadap kebutuhan fungsional dengan menggunakan metode *black box*. Metode *black box* merupakan metode pengujian yang fokus pada pola masukan dan keluaran yang sesuai dengan

skenario. Uji coba ini dilakukan dengan mengacu pada kasus penggunaan yang dijelaskan pada bab 3.

### 5.2.2.1 Uji Coba Melakukan Pencarian Indekos

Pada uji coba ini partisipan akan melakukan pencarian indekos pada aplikasi CariKos. Skenario uji coba pencarian indekos dapat dilihat pada Tabel 5.4 dan hasil uji coba dapat dilihat di Gambar 4.3.

**Tabel 5.4 Skenario Uji Coba Melakukan Pencarian Indekos**

ID	UJ-001
Kasus Penggunaan	Melakukan Pencarian Indekos
Sub Kasus	-
Nama	Pengujian melakukan pencarian indekos
Tujuan Uji Coba	Menguji fitur pencarian indekos
Skenario	Partisipan melakukan pencarian indekos
Kondisi Awal	Partisipan berada di halaman utama
Langkah Uji Coba	<ol style="list-style-type: none"> <li>1. Partisipan membuka halaman utama</li> <li>2. Partisipan menekan tombol Cari Kos</li> <li>3. Partisipan memasukkan kriteria yang ingin dicari</li> <li>4. Partisipan menekan tombol Cari</li> </ol>
Hasil yang Diharapkan	Menampilkan indekos yang sesuai dengan kriteria yang dipilih
Hasil yang Didapat	Menampilkan indekos yang sesuai dengan kriteria yang dipilih
Hasil Uji Coba	Berhasil

### 5.2.2.2 Uji Coba Melakukan Pemesanan Indekos

Pada uji coba ini partisipan akan melakukan pemesanan indekos pada aplikasi CariKos. Skenario uji coba melakukan

pemesanan indekos dapat dilihat pada Tabel 5.5 dan hasil uji coba dapat dilihat di Gambar 4.6.

**Tabel 5.5 Skenario Uji Coba Melakukan Pemesanan Indekos**

ID	UJ-002
Kasus Penggunaan	Melakukan Pemesanan Indekos
Sub Kasus	-
Nama	Pengujian melakukan pemesanan indekos
Tujuan Uji Coba	Menguji fitur pemesanan indekos
Skenario	Partisipan melakukan pemesanan indekos
Kondisi Awal	Partisipan berada di halaman formulir pemesanan kamar indekos
Langkah Uji Coba	1. Partisipan mengisi formulir pemesanan kamar indekos 2. Partisipan menekan tombol Lajut ke Pemesanan
Hasil yang Diharapkan	Menampilkan informasi detail pemesanan
Hasil yang Didapat	Menampilkan informasi detail pemesanan
Hasil Uji Coba	Berhasil

### 5.2.2.3 Uji Coba Melakukan Pembatalan Pemesanan Indekos

Pada uji coba ini partisipan akan melakukan pembatalan pemesanan indekos pada aplikasi CariKos. Skenario uji coba pembatalan pemesanan indekos dapat dilihat pada Tabel 5.6 dan hasil uji coba dapat dilihat di Gambar 4.9.

**Tabel 5.6 Skenario Uji Coba Melakukan Pembatalan Pemesanan Indekos**

ID	UJ-003
Kasus Penggunaan	Melakukan Pembatalan Pemesanan Indekos
Sub Kasus	-

Nama	Pengujian melakukan pembatalan pemesanan indekos
Tujuan Uji Coba	Menguji fitur pembatalan pemesanan indekos
Skenario	Partisipan melakukan pembatalan pemesanan indekos
Kondisi Awal	Partisipan berada di halaman daftar transaksi
Langkah Uji Coba	1. Partisipan menekan tombol Batal 2. Partisipan menekan tombol Ya saat terdapat peringatan pembatalan
Hasil yang Diharapkan	Pembatalan berhasil dengan tidak bisanya tombol Batal ditekan
Hasil yang Didapat	Pembatalan berhasil dengan tidak bisanya tombol Batal ditekan
Hasil Uji Coba	Berhasil

#### 5.2.2.4 Uji Coba Melakukan Konfirmasi Pembayaran Indekos

Pada uji coba ini partisipan akan melakukan konfirmasi pembayaran indekos pada aplikasi CariKos. Skenario uji coba konfirmasi pembayaran indekos dapat dilihat pada Tabel 5.7 dan hasil uji coba dapat dilihat di Gambar 5.4.

**Tabel 5.7 Skenario Melakukan Konfirmasi Pembayaran Indekos**

ID	UJ-004
Kasus Penggunaan	Melakukan Konfirmasi Pembayaran Indekos
Sub Kasus	-
Nama	Pengujian melakukan konfirmasi pembayaran indekos
Tujuan Uji Coba	Menguji fitur konfirmasi pembayaran indekos
Skenario	Partisipan melakukan konfirmasi pembayaran indekos



Kondisi Awal	Partisipan berada di halaman konfirmasi pembayaran
Langkah Uji Coba	1. Partisipan mengisi formulir konfirmasi pembayaran 2. Partisipan menekan tombol Konfirmasi
Hasil yang Diharapkan	Konfirmasi pembayaran masuk ke dalam daftar transaksi di admin
Hasil yang Didapat	Konfirmasi pembayaran masuk ke dalam daftar transaksi di admin
Hasil Uji Coba	Berhasil

Transaksi

Show 

10

 entries

Search:

ID Transaksi	ID Pemesanan	Tanggal Pembayaran	Total Pembayaran	Bank Asal	Nomor Rekening	Nama Pemilik	Status
27	138	2017-05-27 14:05:45	Rp3,003,896	Bank BNI	527045245342	Lino	<div>Verifikasi</div>

Showing 1 to 1 of 1 entries

Previous

1

Next

Gambar 5.4 Pengujian Konfirmasi Pembayaran Indekos

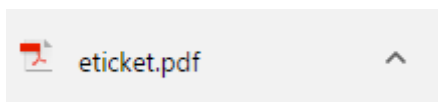
5.2.2.5 Uji Coba Mendapatkan E-Ticket

Pada uji coba ini partisipan akan mendapatkan *e-ticket* dari pemesanan indekos yang sudah dibayar. Skenario uji coba mendapatkan *e-ticket* dapat dilihat pada Tabel 5.8 dan hasil uji coba dapat dilihat di Gambar 5.5.

Tabel 5.8 Skenario Uji Coba Mendapatkan E-Ticket

ID	UJ-005
Kasus Penggunaan	Mendapatkan <i>E-Ticket</i>
Sub Kasus	-
Nama	Pengujian mendapatkan <i>e-ticket</i>

Tujuan Uji Coba	Menguji fitur mendapatkan <i>e-ticket</i>
Skenario	Partisipan mendapatkan <i>e-ticket</i> dari pemesanan yang sudah melakukan konfirmasi pembayaran
Kondisi Awal	Partisipan berada di halaman daftar transaksi
Langkah Uji Coba	1. Partisipan menekan tombol E-Ticket 2. Partisipan menekan tombol ikon download
Hasil yang Diharapkan	Mendapatkan <i>e-ticket</i> dan dapat mengunduh <i>e-ticket</i>
Hasil yang Didapat	Mendapatkan <i>e-ticket</i> dan dapat mengunduh <i>e-ticket</i>
Hasil Uji Coba	Berhasil



**Gambar 5.5 Hasil Uji Coba Mendapatkan E-Ticket**

### 5.2.2.6 Uji Coba Melihat Kamar Indekos

Pada uji coba ini partisipan akan melihat daftar kamar indekos pada setiap indekosnya yang sudah terdaftar di aplikasi CariKos. Skenario uji coba melihat kamar indekos dapat dilihat pada Tabel 5.9 dan hasil uji coba dapat dilihat di Gambar 4.12.

**Tabel 5.9 Skenario Uji Coba Melihat Kamar Indekos**

ID	UJ-006
Kasus Penggunaan	Melihat Kamar Indekos
Sub Kasus	-
Nama	Pengujian melihat kamar indekos
Tujuan Uji Coba	Menguji fitur melihat kamar indekos

Skenario	Partisipan melihat kamar indekos pada setiap indekos yang sudah terdaftar di aplikasi CariKos.
Kondisi Awal	Partisipan berada di halaman utama admin
Langkah Uji Coba	1. Partisipan menekan menu master data 2. Partisipan menekan menu indekos 3. Partisipan menekan nama indekos
Hasil yang Diharapkan	Menampilkan daftar kamar indekos
Hasil yang Didapat	Menampilkan daftar kamar indekos
Hasil Uji Coba	Berhasil

### 5.2.2.7 Uji Coba Melihat Indekos

Pada uji coba ini partisipan akan melihat daftar indekos yang sudah terdaftar di aplikasi CariKos. Skenario uji coba melihat indekos dapat dilihat pada Tabel 5.10 dan hasil uji coba dapat dilihat di Gambar 4.11.

**Tabel 5.10 Skenario Uji Coba Melihat Indekos**

ID	UJ-007
Kasus Penggunaan	Melihat Indekos
Sub Kasus	-
Nama	Pengujian melihat indekos
Tujuan Uji Coba	Menguji fitur melihat indekos
Skenario	Partisipan melihat indekos yang sudah terdaftar di aplikasi CariKos.
Kondisi Awal	Partisipan berada di halaman utama admin
Langkah Uji Coba	1. Partisipan menekan menu master data 2. Partisipan menekan menu indekos
Hasil yang Diharapkan	Menampilkan daftar indekos

Hasil yang Didapat	Menampilkan daftar indekos
Hasil Uji Coba	Berhasil

#### 5.2.2.8 Uji Coba Melihat Transaksi

Pada uji coba ini partisipan akan melihat daftar transaksi di aplikasi CariKos. Skenario uji coba melihat transaksi dapat dilihat pada Tabel 5.11 dan hasil uji coba dapat dilihat di Gambar 4.14.

**Tabel 5.11 Skenario Uji Coba Melihat Transaksi**

ID	UJ-008
Kasus Penggunaan	Melihat Transaksi
Sub Kasus	-
Nama	Pengujian melihat transaksi
Tujuan Uji Coba	Menguji fitur melihat transaksi
Skenario	Partisipan melihat daftar transaksi di aplikasi CariKos.
Kondisi Awal	Partisipan berada di halaman utama admin
Langkah Uji Coba	1. Partisipan menekan menu master data 2. Partisipan menekan menu transaksi
Hasil yang Diharapkan	Menampilkan daftar transaksi
Hasil yang Didapat	Menampilkan daftar transaksi
Hasil Uji Coba	Berhasil

#### 5.2.2.9 Uji Coba Verifikasi Pembayaran

Pada uji coba ini partisipan akan melakukan verifikasi pembayaran. Skenario uji coba melakukan verifikasi pembayaran dapat dilihat pada Tabel 5.11 dan hasil uji coba dapat dilihat di Gambar 4.14.

ID	UJ-009
Kasus Penggunaan	Verifikasi Pembayaran
Sub Kasus	-
Nama	Pengujian melakukan verifikasi pembayaran
Tujuan Uji Coba	Menguji fitur melakukan verifikasi pembayaran
Skenario	Partisipan melakukan verifikasi pembayaran
Kondisi Awal	Partisipan berada di halaman utama admin
Langkah Uji Coba	1. Partisipan menekan menu master data 2. Partisipan menekan menu transaksi 3. Partisipan menekan tombol Verifikasi
Hasil yang Diharapkan	Status berubah menjadi lunas
Hasil yang Didapat	Status berubah menjadi lunas
Hasil Uji Coba	Berhasil

## Transaksi

Show 

10

 entries

Search:

ID Transaksi ▲	ID Pemesanan ◂	Tanggal Pembayaran ◂	Total Pembayaran ◂	Bank Asal ◂	Nomor Rekening ◂	Nama Pemilik ◂	Status ◂
27	138	2017-05-27 14:55:08	Rp3,003,896	Bank BNI	527045245342	Lino	Lunas

Showing 1 to 1 of 1 entries

Previous

1

Next

**Gambar 5.6 Hasil Uji Coba Verifikasi Pembayaran**

### 5.2.2.10 Uji Coba Melihat Laporan Keuangan

Pada uji coba ini partisipan akan melihat laporan keuangan dari aplikasi CariKos. Skenario uji coba melihat laporan keuangan dapat dilihat pada Tabel 5.12 dan hasil uji coba dapat dilihat di Gambar 4.15.

**Tabel 5.12 Skenario Uji Coba Melihat Laporan Keuangan**

ID	UJ-010
Kasus Penggunaan	Melihat Laporan Keuangan
Sub Kasus	-
Nama	Pengujian melihat laporan keuangan
Tujuan Uji Coba	Menguji fitur melihat laporan keuangan
Skenario	Partisipan melihat laporan keuangan
Kondisi Awal	Partisipan berada di halaman utama admin
Langkah Uji Coba	<ol style="list-style-type: none"> <li>1. Partisipan menekan menu laporan keuangan</li> <li>2. Partisipan memasukkan bulan dan tahun yang akan dilihat</li> <li>3. Partisipan menekan tombol Pilih</li> </ol>
Hasil yang Diharapkan	Menampilkan laporan keuangan
Hasil yang Didapat	Menampilkan laporan keuangan
Hasil Uji Coba	Berhasil

#### 5.2.2.11 Uji Coba Melihat Pemesanan

Pada uji coba ini partisipan akan melihat daftar pemesanan di aplikasi CariKos. Skenario uji coba melihat daftar pemesanan dapat dilihat pada Tabel 5.13 dan hasil uji coba dapat dilihat di Gambar 4.16.

**Tabel 5.13 Skenario Uji Coba Melihat Pemesanan**

ID	UJ-011
Kasus Penggunaan	Melihat Pemesanan
Sub Kasus	-
Nama	Pengujian melihat pemesanan

Tujuan Uji Coba	Menguji fitur melihat pemesanan
Skenario	Partisipan melihat pemesanan
Kondisi Awal	Partisipan berada di halaman utama admin
Langkah Uji Coba	1. Partisipan menekan menu master data 2. Partisipan menekan menu reservasi
Hasil yang Diharapkan	Menampilkan daftar pemesanan
Hasil yang Didapat	Menampilkan daftar pemesanan
Hasil Uji Coba	Berhasil

5.2.3 Skenario Uji Coba Penentuan Harga

Pada uji coba penentuan harga ini akan dilihat perbandingan harga awal dengan penentuan harga yang sudah dilakukan menggunakan metode back propagation. Gambar 5.7 merupakan harga awal dari kamar indekos, Gambar 5.8 merupakan penentuan harga jika permintaan sedang rendah, dan Gambar 5.9 merupakan penentuan harga jika permintaan sedang tinggi. Untuk itu dipilih bulan Februari sebagai bulan yang memiliki permintaan rendah dan bulan Agustus sebagai bulan yang memiliki permintaan yang banyak. Daftar permintaan dapat dilihat pada Gambar 1.1.

Daftar Kamar Kos Trompet

Jenis Kamar	Harga Kamar	Jumlah Kamar	Luas Kamar (Dalam m)	
Kamar Deluxe	1,500,000	6	6 X 4	Hapus

Gambar 5.7 Harga Awal Kamar Deluxe Indekos Terompet



Kamar Deluxe

---

Kos Trompet  
Rp1,471,841/bulan  
Tersedia 5 Kamar

Lihat Kamar

**Gambar 5.8 Harga Akhir Ketika Permintaan Rendah**



Kamar Deluxe

---

Kos Trompet  
Rp1,578,354/bulan  
Tersedia 5 Kamar

Lihat Kamar

**Gambar 5.9 Harga Akhir Ketika Permintaan Tinggi**



### 5.2.4 Skenario Uji Coba Tingkat Kepuasan serta Pengukuran Keuntungan atau Kerugian

Pada uji coba tingkat kepuasan ini akan dilakukan pemesanan indeks dalam satu periode oleh lima belas partisipan. Setiap partisipan akan memesan indeks melalui aplikasi CariKos. Pemesanan dilakukan tiga kali dalam satu periode. Waktu untuk pemesanan sudah ditentukan. Daftar partisipan dapat dilihat pada Tabel 5.14. Penentuan waktu dapat dilihat pada Tabel 5.15.

**Tabel 5.14 Daftar Partisipan**

No	Nama	Pekerjaan
1	M. Irfan Andhikaputra	Mahasiswa
2	Glleen Allan M.	Mahasiswa
3	Hamka Aminullah	Mahasiswa
4	Fikry Khairytamim	Mahasiswa
5	Tiara Anggita	Mahasiswa
6	Bambang A. Yudhistira	Mahasiswa
7	Bayu Rizky Ramadhan	Mahasiswa
8	Annisa Ziarieputi	Mahasiswa
9	Rifki Lufthansa	Mahasiswa
10	Abi Nubli	Mahasiswa
11	Ekaprana Danian	Mahasiswa
12	Chandra Mraz	Mahasiswa
13	Yuan Dewi Florean	Calon Mahasiswa
14	Yonathan Edho	Calon Mahasiswa
15	Yoel Ivan Pratama	Calon Mahasiswa

**Tabel 5.15 Penentuan Waktu pada Uji Coba**

Partisipan	Percobaan ke -		
	1	2	3
1	Februari	Desember	Agustus
2	Juli	Februari	November
3	November	Februari	Agustus

4	Agustus	Juni	November
5	Mei	September	Februari
6	Agustus	Desember	Januari
7	September	Maret	Juni
8	Agustus	Februari	Mei
9	Agustus	Desember	Februari
10	November	Januari	Juni
11	September	Maret	Juni
12	Desember	Mei	Juli
13	September	Februari	Juni
14	November	Maret	Juli
15	Oktober	April	Agustus

Setelah menentukan waktu, partisipan memilih indekos yang sesuai kriteria sesuai waktu yang dipilih. Harga indekos yang dipilih oleh partisipan dan harga dasar dari indekos dapat dilihat pada Tabel 5.16.

**Tabel 5.16 Harga Indekos yang Dipilih Partisipan pada Uji Coba**

Partisipan	Percobaan ke -	Harga Dasar	Harga Akhir
1	1	Rp1.500.000	Rp1.486.978
	2	Rp1.400.000	Rp1.365.403
	3	Rp1.350.000	Rp1.419.148
2	1	Rp750.000	Rp787.528
	2	Rp800.000	Rp778.293
	3	Rp900.000	Rp875.056
3	1	Rp900.000	Rp875.056
	2	Rp900.000	Rp875.056
	3	Rp900.000	Rp945.764
4	1	Rp1.500.000	Rp1.577.110
	2	Rp1.500.000	Rp1.564.728
	3	Rp1.500.000	Rp1.461.815
5	1	Rp1.400.000	Rp1.421.974
	2	Rp1.400.000	Rp1.466.095

	3	Rp1.400.000	Rp1.363.129
6	1	Rp1.350.000	Rp1.419.148
	2	Rp1.350.000	Rp1.316.638
	3	Rp1.350.000	Rp1.333.953
7	1	Rp750.000	Rp785.948
	2	Rp750.000	Rp734.592
	3	Rp750.000	Rp782.760
8	1	Rp1.500.000	Rp1.577.110
	2	Rp1.500.000	Rp1.461.815
	3	Rp1.500.000	Rp1.526.652
9	1	Rp1.350.000	Rp1.419.148
	2	Rp1.350.000	Rp1.316.638
	3	Rp1.350.000	Rp1.314.446
10	1	Rp900.000	Rp875.580
	2	Rp900.000	Rp887.644
	3	Rp900.000	Rp937.512
11	1	Rp800.000	Rp837.412
	2	Rp800.000	Rp780.871
	3	Rp800.000	Rp833.344
12	1	Rp1.350.000	Rp1.316.638
	2	Rp1.350.000	Rp1.371.189
	3	Rp1.350.000	Rp1.416.901
13	1	Rp900.000	Rp942.089
	2	Rp900.000	Rp875.580
	3	Rp900.000	Rp937.512
14	1	Rp1.500.000	Rp1.461.815
	2	Rp1.500.000	Rp1.467.619
	3	Rp1.500.000	Rp1.574.748
15	1	Rp750.000	Rp788.660
	2	Rp750.000	Rp731.483
	3	Rp750.000	Rp764.528

Dari Tabel 5.16 dapat dilihat bahwa selisih harga dasar dengan harga akhir dibawah Rp80,000. Harga akhir merupakan

harga yang didapat dari penentuan harga dengan metode *back propagation*. Dari selisih ini partisipan menganggap kenaikan harga ketika permintaan tinggi masih dibatas wajar dan tidak terlalu mahal. Kepuasan partisipan dapat dilihat pada Tabel 5.17. Kepuasan tersebut didapatkan dari pertanyaan pada Gambar A.2. Dapat dilihat pada Tabel 5.17. Selain kepuasan partisipan dapat dilihat juga keuntungan/kerugian yang dialami oleh aplikasi CariKos selama satu periode. Hasil keuntungan/kerugian dapat dilihat pada Tabel 5.18.

**Tabel 5.17 Hasil Uji Coba Tingkat Kepuasan Terhadap Harga Baru**

Partisipan	Tingkat Kepuasan Terhadap Harga
1	Puas
2	Puas
3	Puas
4	Puas
5	Puas
6	Puas
7	Puas
8	Puas
9	Puas
10	Puas
11	Puas
12	Puas
13	Puas
14	Puas
15	Puas

**Tabel 5.18 Hasil Uji Coba Pengukuran Keuntungan atau Kerugian**

Partisipan	Total Keuntungan atau Kerugian
1	Rp21.529
2	- Rp24.944
3	- Rp4.123
4	Rp103.653
5	Rp51.198

6	Rp19.739
7	Rp53.300
8	Rp65.577
9	Rp12.438
10	Rp736
11	Rp51.627
12	Rp54.728
13	Rp55.181
14	Rp4.182
15	Rp34.671
Total	Rp499.492

Dari Tabel 5.18 dapat dilihat bahwa aplikasi CariKos dalam melakukan skenario mendapatkan keuntungan sebesar Rp499.492.

### 5.3 Evaluasi Uji Coba

Pada bagian ini akan dijelaskan hasil evaluasi dari uji coba yang telah dilakukan sebelumnya. Evaluasi yang diberikan meliputi evaluasi dari uji coba back propagation sebelumnya, evaluasi uji coba fungsionalitas sebelumnya, dan evaluasi uji coba tingkat kepuasan serta pengukuran keuntungan dan kerugian sebelumnya.

#### 5.3.1 Evaluasi Uji Coba Back Propagation

Hasil uji coba back propagation mendapatkan hasil *learning rate* bernilai 0.9, *hidden layer* berjumlah tiga, dan iterasi sebanyak 180 kali merupakan hasil yang paling optimal dikarenakan menghasilkan nilai MSE yang kecil. Maka dari itu hasil tersebut digunakan dalam *training* data untuk mendapatkan *weight* dari per *node*. Setelah itu melakukan penentuan harga dengan menggunakan *weight* yang sudah di dapatkan. Hasil evaluasi dari uji coba back propagation dapat dilihat pada Tabel 5.18.

**Tabel 5.19 Hasil Evaluasi Uji Coba Back Propagation**

Data	Input		Target	Hasil <i>Output</i>	MSE
	Permintaan	Skor			
1	0.21	0.3	0	0.019	0.000361
2	0.85	0.42	1	0.811	0.035721
3	0.74	0.58	1	0.819	0.032761
4	0.1	0.22	0	0.007	0.000049
5	0.42	0.26	0	0.067	0.004489
6	0.14	0.58	0	0.057	0.003249
7	0.52	0.34	0	0.218	0.047524
8	0.15	0.46	0	0.023	0.000529
9	0.41	0.74	1	0.585	0.172225
10	0.1	0.66	0	0.069	0.004761
Rata-rata MSE					0.030167

Dari hasil evaluasi pada Tabel 5.18 didapatkan nilai MSE sebesar 0.030167. Dapat dilihat bahwa MSE yang didapatkan dari back propagation yang digunakan pada aplikasi ini kecil, sehingga penentuan harga yang dihasilkan optimal. Nilai MSE yang dihasilkan kecil dikarena nilai *input* tidak terlalu besar dan beragam.

### 5.3.2 Evaluasi Uji Coba Fungsionalitas

Rangkuman mengenai hasil uji coba fungsionalitas yang dilakukan pada 5 partisipan yang menjalankan aplikasi CariKos sebagai pencari indekos dapat dilihat pada Tabel 5.19.

**Tabel 5.20 Evaluasi Uji Coba Fungsional Pencari Indekos**

No	ID	Kebutuhan Fungsional	Terpenuhi
1	UJ-001	Uji coba melakukan pencarian indekos	5

2	UJ-002	Uji coba melakukan pemesanan indekos	5
3	UJ-003	Uji coba melakukan pembatalan pemesanan indekos	5
4	UJ-004	Uji coba melakukan konfirmasi pembayaran	5
5	UJ-005	Uji coba mendapatkan <i>e-ticket</i>	5

Sedangkan hasil uji coba fungsionalitas secara keseluruhan dapat dilihat pada Tabel 5.20. Dapat dilihat pada tabel tersebut, seluruh skenario uji coba yang telah dilakukan berhasil dan aplikasi berjalan dengan baik, sehingga dapat ditarik kesimpulan bahwa fungsionalitas dari aplikasi bekerja sesuai dengan yang diharapkan.

**Tabel 5.21 Evaluasi Uji Coba Fungsionalitas**

No	ID	Kebutuhan Fungsional	Hasil
1	UJ-001	Uji coba melakukan pencarian indekos	Berhasil
2	UJ-002	Uji coba melakukan pemesanan indekos	Berhasil
3	UJ-003	Uji coba melakukan pembatalan pemesanan indekos	Berhasil
4	UJ-004	Uji coba melakukan konfirmasi pembayaran	Berhasil
5	UJ-005	Uji coba mendapatkan <i>e-ticket</i>	Berhasil
6	UJ-006	Uji coba melihat kamar indekos	Berhasil
7	UJ-007	Uji coba melihat indekos	Berhasil
8	UJ-008	Uji coba melihat transaksi	Berhasil
9	UJ-009	Uji coba verifikasi pembayaran	Berhasil
10	UJ-010	Uji coba melihat laporan keuangan	Berhasil

11	UJ-011	Uji coba melihat pemesanan	Berhasil
----	--------	----------------------------	----------

### 5.3.3 Evaluasi Uji Coba Tingkat Kepuasan serta Pengukuran Keuntungan atau Kerugian

Rangkuman hasil uji coba tingkat kepuasan dari lima belas partisipan dalam melakukan tiga kali pemesanan dalam satu periode dapat dilihat pada Tabel 5.17 dan Gambar 5.10.



**Gambar 5.10 Grafik Tingkat Kepuasan**

Berdasarkan Gambar 5.10 dapat ditarik kesimpulan bahwa semua partisipan merasa puas dengan harga yang ditawarkan oleh aplikasi CariKos yang merupakan harga dari perhitungan metode back propagation. Hal ini dikarenakan menurut semua partisipan dengan adanya aplikasi CariKos ini dapat membantu para pencari indekos, walau saat waktu permintaan sedang tinggi harga yang ditawarkan lebih mahal dari pada harga yang sebenarnya. Harga yang lebih mahal itu masih dianggap wajar.

Pada aplikasi CariKos harga yang ditawarkan *fluktuatif* sesuai dengan permintaan pasar. Ketika permintaan sedang rendah, harga yang ditawarkan akan lebih murah dari harga sebenarnya dan



ketika permintaan sedang tinggi harga yang ditawarkan lebih mahal. Walaupun begitu dari uji coba yang dilakukan pada Tabel 5.15 dan Tabel 5.17 terlihat bahwa aplikasi CariKos mengalami keuntungan.

#### **5.3.4 Evaluasi Uji Coba Penentuan Harga**

Dari hasil uji coba penentuan harga, didapatkan bahwa ketika permintaan sedang tinggi maka harga mengalami kenaikan. Sedangkan ketika permintaan sedang rendah maka harga mengalami penurunan. Hasil uji coba dapat dilihat pada Gambar 5.7, Gambar 5.8, dan Gambar 5.9.

*[Halaman ini sengaja dikosongkan]*

## **BAB VI**

### **KESIMPULAN DAN SARAN**

Pada bab ini akan dibahas tentang kesimpulan yang diambil selama pengerjaan Tugas Akhir beserta saran-saran tentang pengembangannya yang dapat dilakukan terhadap Tugas Akhir ini di masa yang akan datang.

#### **6.1 Kesimpulan**

Selama proses perancangan, implementasi, dan uji coba dapat diambil kesimpulan sebagai berikut:

1. Implementasi back propagation dengan mempertimbangkan permintaan dan nilai skor indekos yang sudah dinormalisasi.
2. Untuk mendapatkan eror yang kecil, dilakukan uji coba penentuan *learning rate*, jumlah *hidden layer*, dan jumlah iterasi agar menghasilkan nilai eror yang kecil.
3. Back propagation yang digunakan untuk menentukan harga indekos menghasilkan nilai rata-rata MSE sebesar 0.030167.
4. Penentuan harga dengan mendenormalisasikan hasil *output* dari back propagation.
5. Aplikasi CariKos dapat melakukan pencarian indekos di sekitar Institut Teknologi Sepuluh Nopember Surabaya.
6. Proses bisnis yang terjadi di aplikasi CariKos dalam mencari, memesan, dan membayar indekos adalah sebagai berikut:
  - a. Pencari indekos mencari indekos yang sesuai dengan kriteria.
  - b. Pencari indekos memesan indekos dari hasil pencarian.
  - c. Bila pencari indekos ingin membatalkan pemesanan, pencari indekos dapat membatalkan pemesanan indekos.
  - d. Pencari indekos melakukan konfirmasi pembayaran.
  - e. Admin melakukan verifikasi pembayaran.
  - f. Pencari indekos mendapatkan *e-ticket* dari indekos yang sudah dibayar.

## 6.2 Saran

Saran yang diberikan terkait pengembangan pada Tugas Akhir ini adalah:

1. Jika pemesanan lebih dari satu bulan, diharapkan kedepannya harganya juga mengikuti sesuai dengan permintaan.
2. Bisa ditambahkan fitur *QR-Code* didalam *e-ticket* sehingga dapat mempermudah pemilik indekos melakukan konfirmasi pemesanan indekos.
3. Aplikasi CariKos dapat diaplikasikan ke aplikasi perangkat bergerak pada *platform* android dan ios.
4. Menambahkan fitur *payment gateway* agar mempermudah melakukan konfirmasi pembayaran.

## DAFTAR PUSTAKA

- [1] D. Irmawati, “Pemanfaatan E-Commers Dalam Dunia Bisnis,” *J. Ilm. Orasi Bisnis*, vol. 6, hal. 95–112, Nov 2011.
- [2] Liputan6.com, “Pertumbuhan Pasar E-Commerce Indonesia Lampauai Amerika Serikat,” *liputan6.com*. [Daring]. Tersedia pada: <http://tekno.liputan6.com/read/631288/pertumbuhan-pasar-e-commerce-indonesia-lampauai-amerika-serikat>. [Diakses: 04-Mei-2017].
- [3] E. Christianto, “Faktor yang mempengaruhi volume impor beras di Indonesia,” *J. JIBEKA*, vol. 7, hal. 38–43, Agustus 2013.
- [4] E. Yohannes, W. F. Mahmudy, dan A. Rahmi, “Penentuan Upah Minimum Kota Berdasarkan Tingkat Inflasi Menggunakan Backpropagation Neural Network (BPNN),” *J. Teknol. Inf. Dan Ilmu Komput.*, vol. 2, no. 1, hal. 34–40, Des 2015.
- [5] “11 Best PHP Frameworks for Modern Web Developers in 2017,” *Coders Eye - Web Dev Tutorials and How-To Guides for Beginners*, 28-Sep-2016. .
- [6] “Sublime Text,” *Wikipedia*. 12-Apr-2017.
- [7] “PHP,” *Wikipedia*. 27-Apr-2017.

*[Halaman ini sengaja dikosongkan]*

## **LAMPIRAN**

1. Pada bulan apa Anda biasanya melakukan pemesanan indekos?

- ☐ Januari
- ☐ Februari
- ☐ Maret
- ☐ April
- ☐ Maret
- ☐ Mei
- ☐ Juni
- ☐ Juli
- ☐ Agustus
- ☐ September
- ☐ Oktober
- ☐ November
- ☐ Desember

**Gambar A.1 Pertanyaan Kuesioner Permintaan Pasar**

1. Seberapa puaskah Anda terhadap harga yang ditampilkan di aplikasi CariKos?

- ☐ Sangat puas
- ☐ Puas
- ☐ Tidak puas
- ☐ Sangat tidak puas

**Gambar A.2 Pertanyaan Kuesioner Tingkat Kepuasan Terhadap Harga yang Ditampilkan**

*[Halaman ini sengaja dikosongkan]*



## BIODATA PENULIS



Rifqi Nur Fadhilah, lahir di Serang pada tanggal 11 Maret 1995. Penulis menempuh pendidikan mulai dari TK Kharisma Serdang Serang (2001), SDN 1 Cilegon (2001-2007), SMPN 1 Cilegon (2007-2010), SMAN 2 Krakatau Steel Cilegon (2010-2013), dan sekarang sedang menjalani pendidikan S1 Teknik Informatika di ITS. Penulis aktif dalam organisasi Himpunan Mahasiswa Teknik Computer (HMTC). Diantaranya adalah menjadi staff magang departemen kesejahteraan mahasiswa himpunan mahasiswa teknik computer ITS 2013-2014. Penulis juga aktif dalam kegiatan kepanitiaan Schematics. Diantaranya penulis pernah menjadi staff keamanan dan transportasi Schematics 2014 dan Badan Pengurus Harian (BPH) perlengkapan dan transportasi 2015. Penulis mengambil bidang minat Manajemen Informasi (MI). Komunikasi dengan penulis dapat melalui email: **rifqinurfadhilah@yahoo.com**